POD-based reduction methods, the Quasi-continuum method and their resemblance

Lars Beex, Stéphane Bordas

Elisa Schenone, Jack S. Hale

Ron Peerlings, Marc Geers

Pierre Kerfriden

RUES | RESEARCH UNIT IN ENGINEERING SCIENCES

UNIVERSITÉ DU LUXEMBOURG

FACULTY OF SCIENCES, TECHNOLOGY AND COMMUNICATION

Aim

Large nonlinear models are inefficient due to:

1. **Many DOFs**

2. **Many integration points**

Aim

Large nonlinear models are inefficient due to:

1. **Many DOFs** — **Interpolation**
2. **Many integration points** — **Reduced integration**

Outline

1. Applications: discrete materials & discrete models

Outline

1. Applications: discrete materials & discrete models

2. The Quasicontinuum method:

   interpolation & integration

Outline

1. Applications: discrete materials & discrete models

2. The Quasicontinuum method:

   interpolation & integration

3. POD-based reduction methods:

   interpolation & integration

Outline

1. Applications: discrete materials & discrete models

2. The Quasicontinuum method:

   interpolation & integration

3. POD-based reduction methods:

   interpolation & integration

4. Concluding remarks

Outline

1. Applications: discrete materials & discrete models

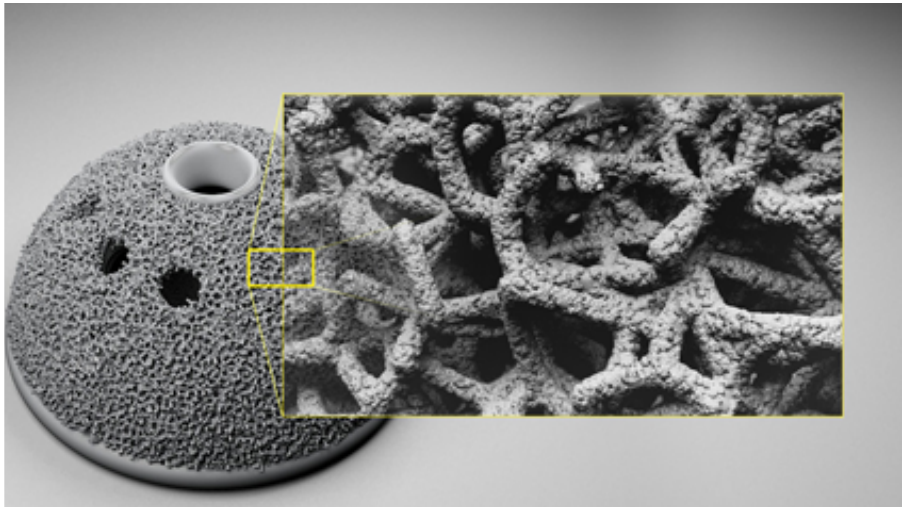
2. The Quasicontinuum method:
   interpolation & integration

3. POD-based reduction methods:
   interpolation & integration

4. Concluding remarks

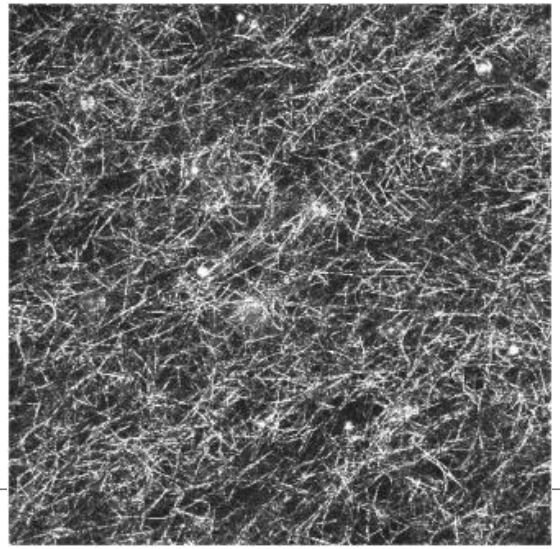
Some discrete materials

Foams

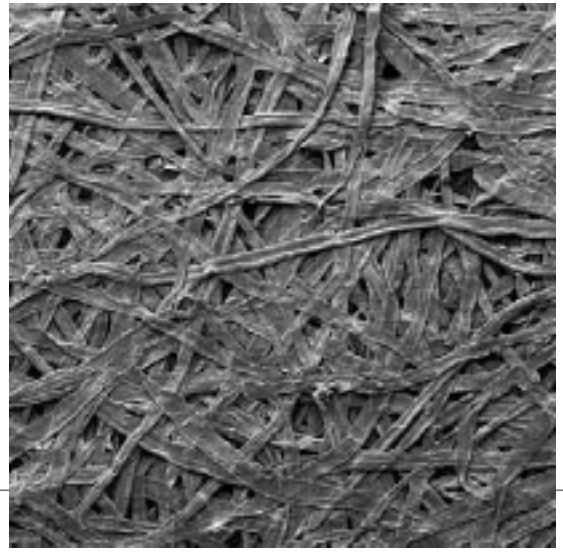


Additive manufacturing

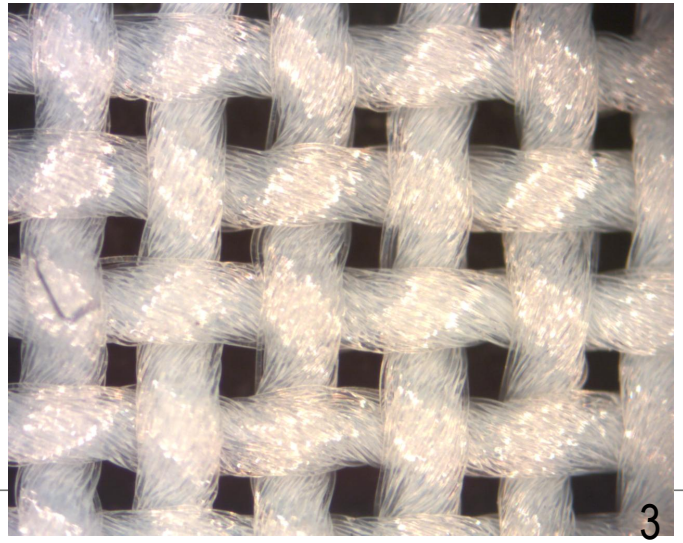
Collagen

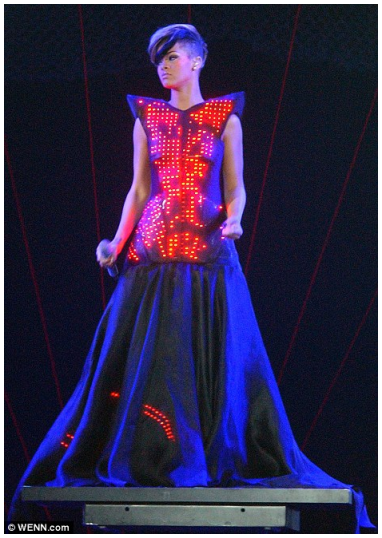


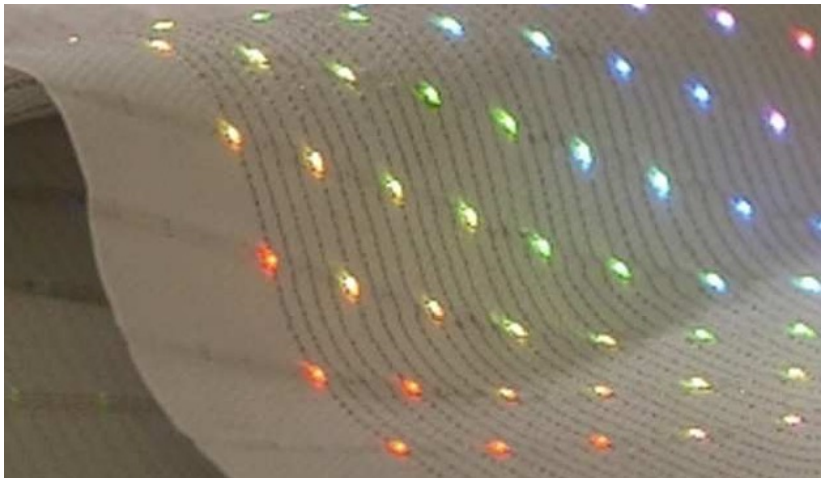
Paper/cardboard

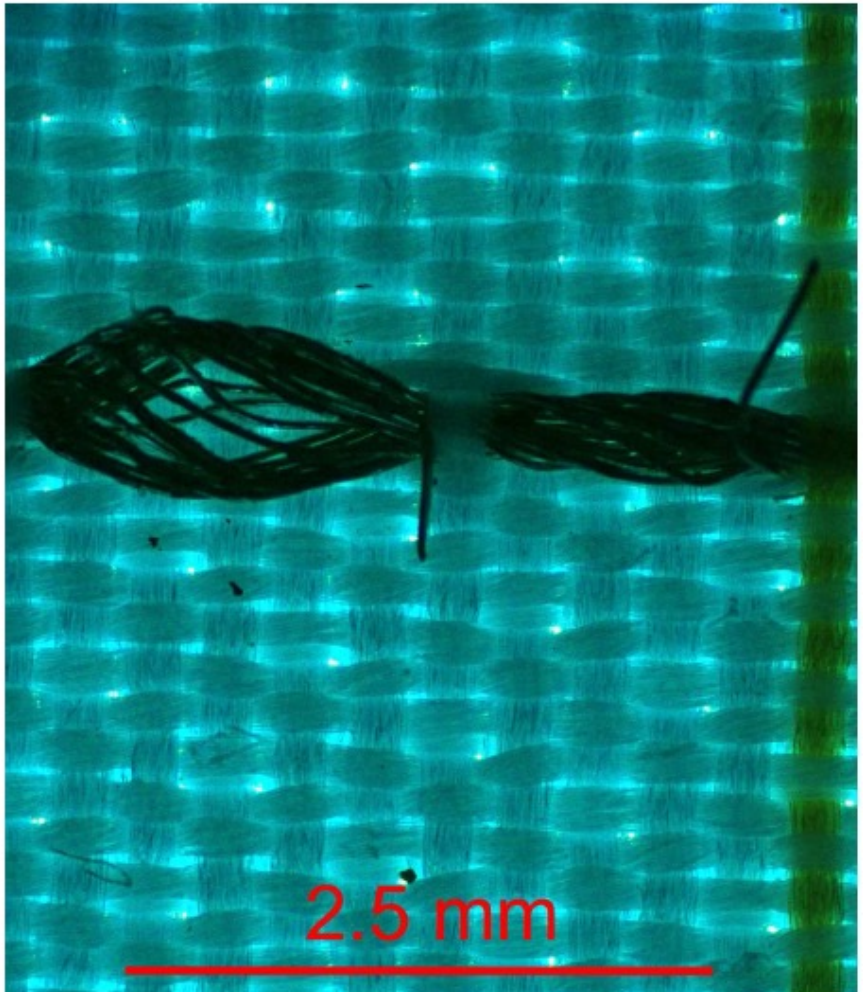


Textiles

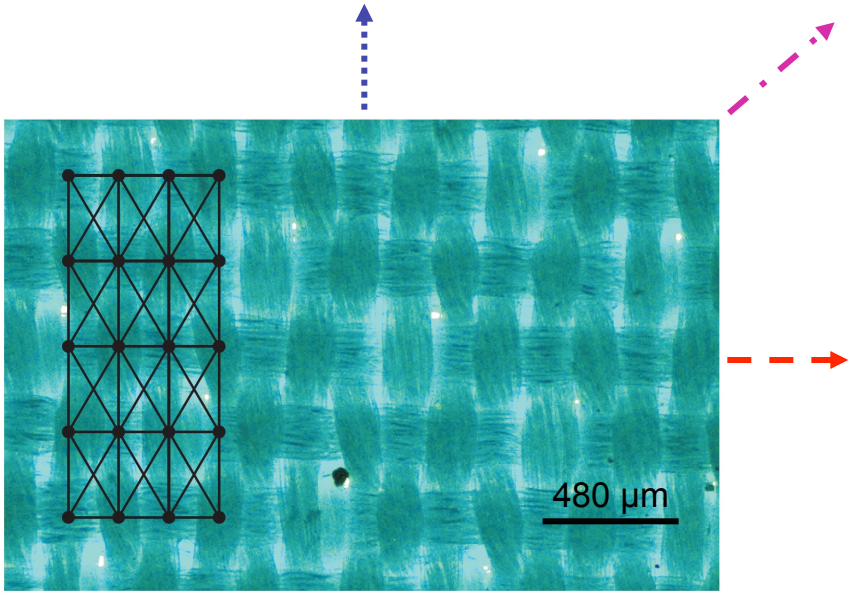


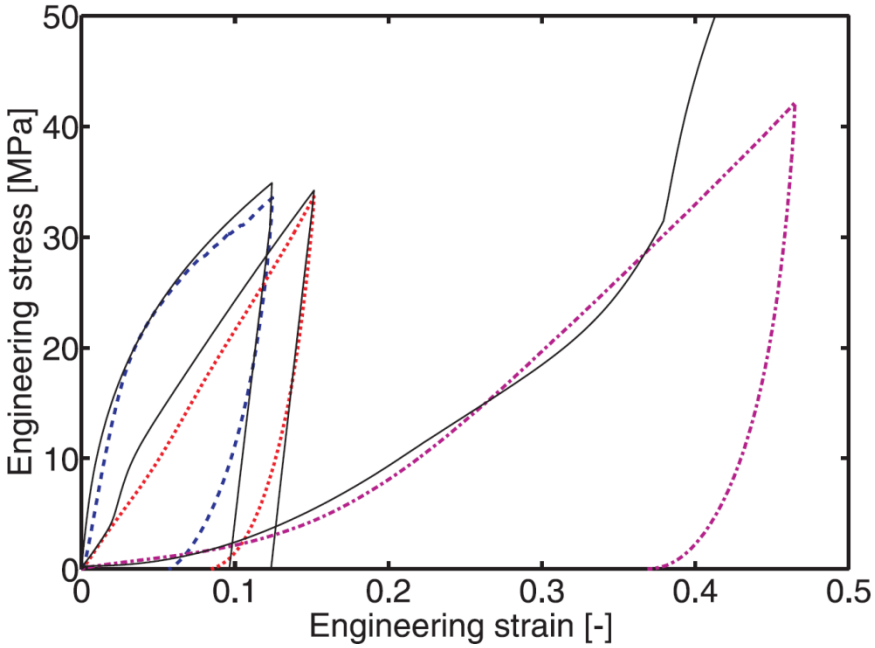

© WENN.com




2.5 mm

Elastoplastic spring lattice





Outline

1. Applications: discrete materials & discrete models

2. The Quasicontinuum method:

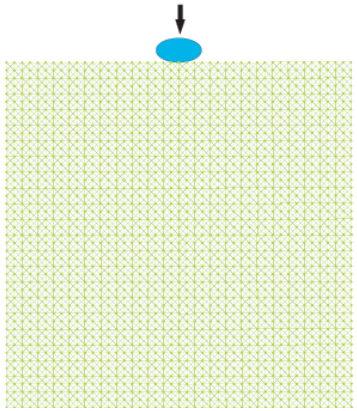
interpolation & integration

3. POD-based reduction methods:

interpolation & integration

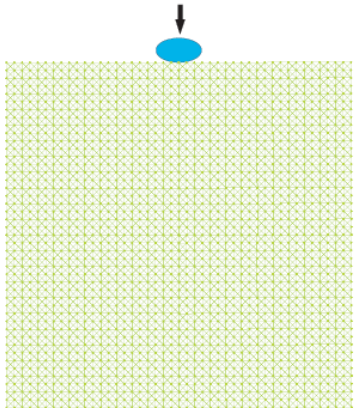
4. Concluding remarks

Direct simulation of elastic lattice



$$E_{tot}(u) = E_{int}(u) - f_{ex}^{T} u$$

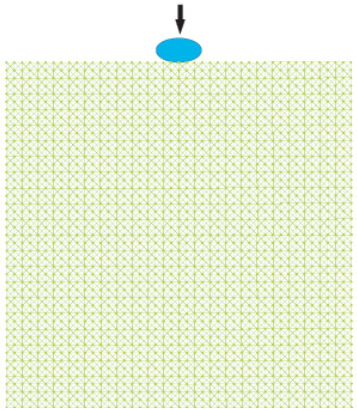
Direct simulation of elastic lattice



$$E_{tot}(u) = E_{int}(u) - f_{ex}^{T} u$$

$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^{T} u$$

Interpolation

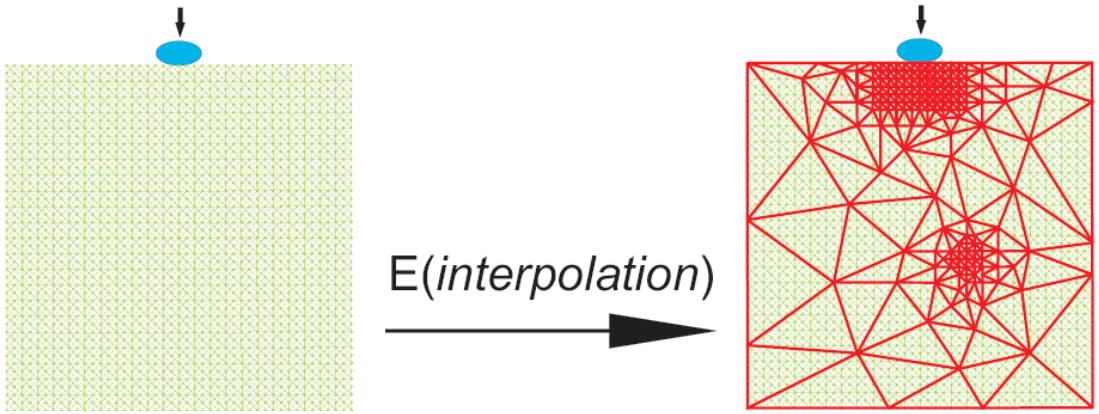


$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^T u \qquad E_{tot}(Nu_t) = \sum_{i=1}^{n} E_i(Nu_t) - f_{ex}^T Nu_t$$

$$u = Nu_t$$

Quasicontinuum method (Tadmor et al., 1996)

Linear interpolation



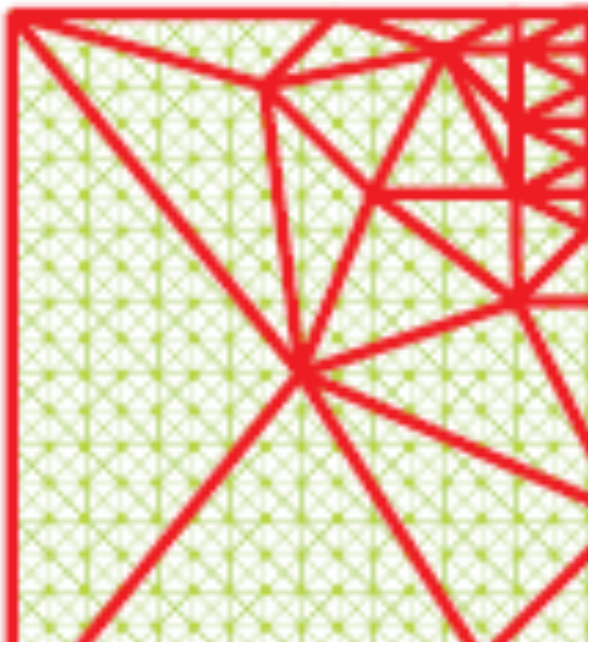
$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^T u \qquad E_{tot}(Nu_t) = \sum_{i=1}^{n} E_i(Nu_t) - f_{ex}^T Nu_t$$

$$u = Nu_t$$

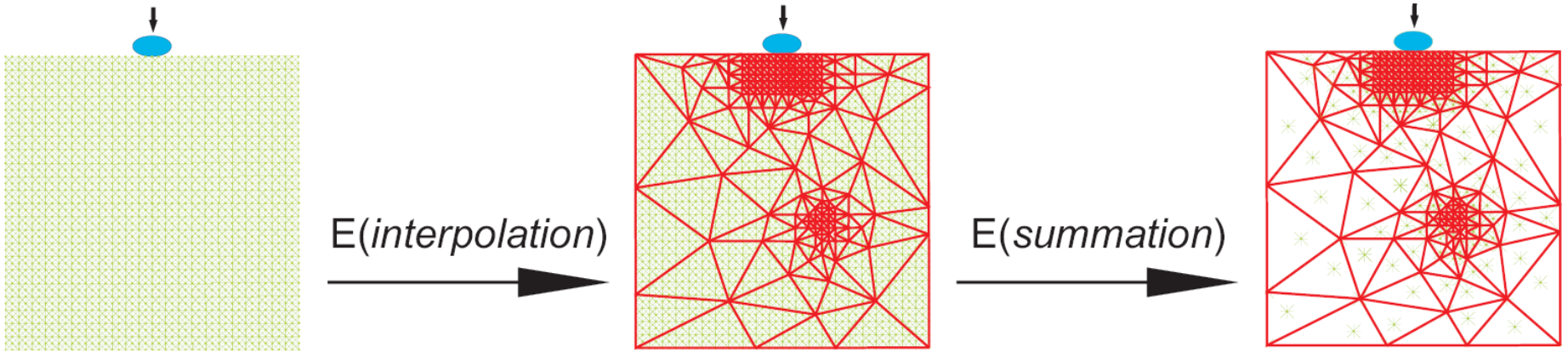
Quasicontinuum method (Tadmor et al., 1996)

Linear interpolation

$$u = Nu_t$$



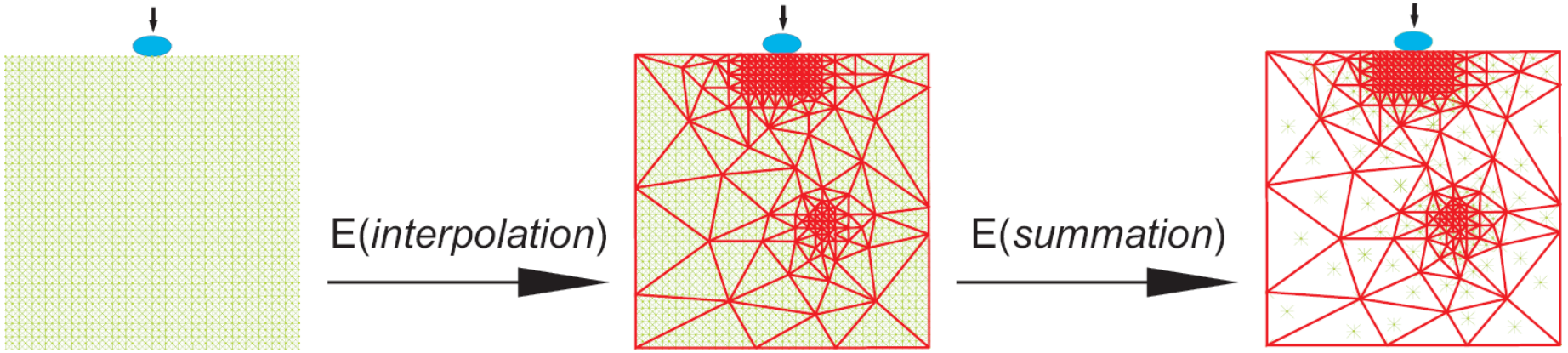
Integration for linear triangles



$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^T u \qquad E_{tot}(Nu_t) = \sum_{i=1}^{n} E_i(Nu_t) - f_{ex}^T Nu_t$$

$$u = Nu_t$$

Integration for linear triangles



$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^T u$$

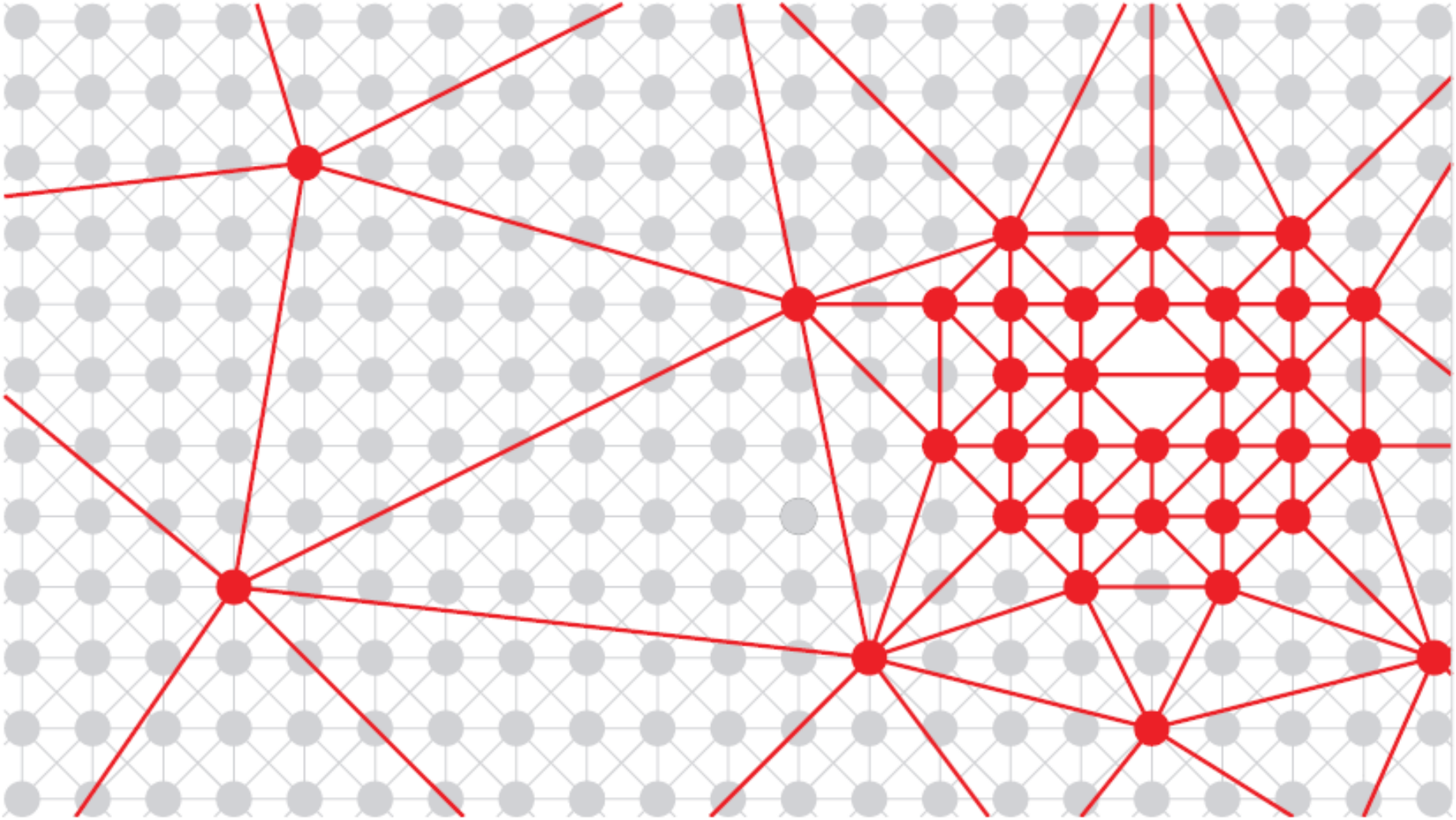
$$E_{tot}(Nu_t) = \sum_{i=1}^{n} E_i(Nu_t) - f_{ex}^T Nu_t$$

$$E_{tot}(Nu_t) = \sum_{i=1}^{s} E_i(Nu_t) - f_{ex}^T Nu_t$$

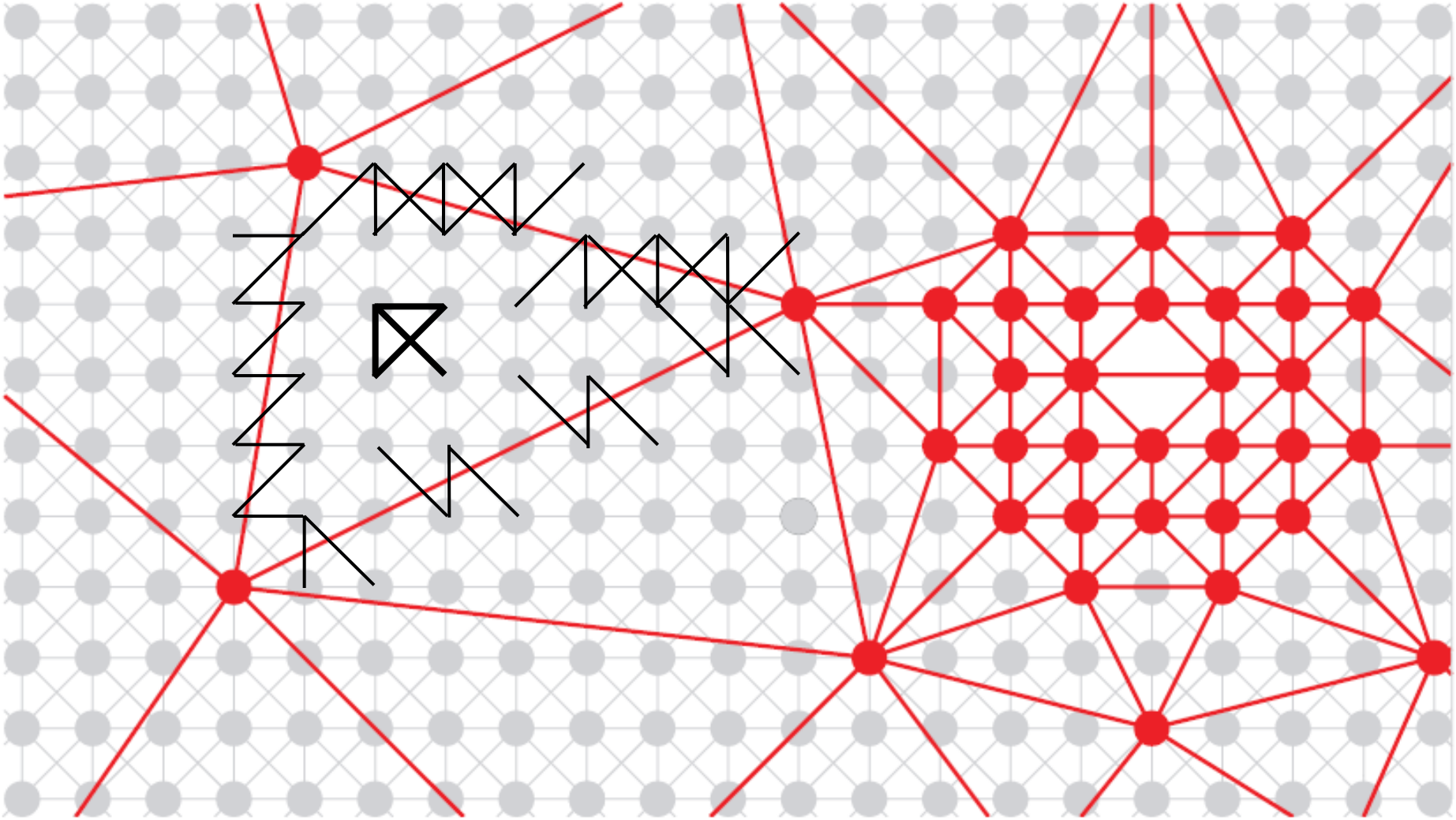
$$u = Nu_t$$

$$\sum_{i=1}^{n} E_i(Nu_t) \approx \sum_{i=1}^{s} w_i E_i(Nu_t)$$

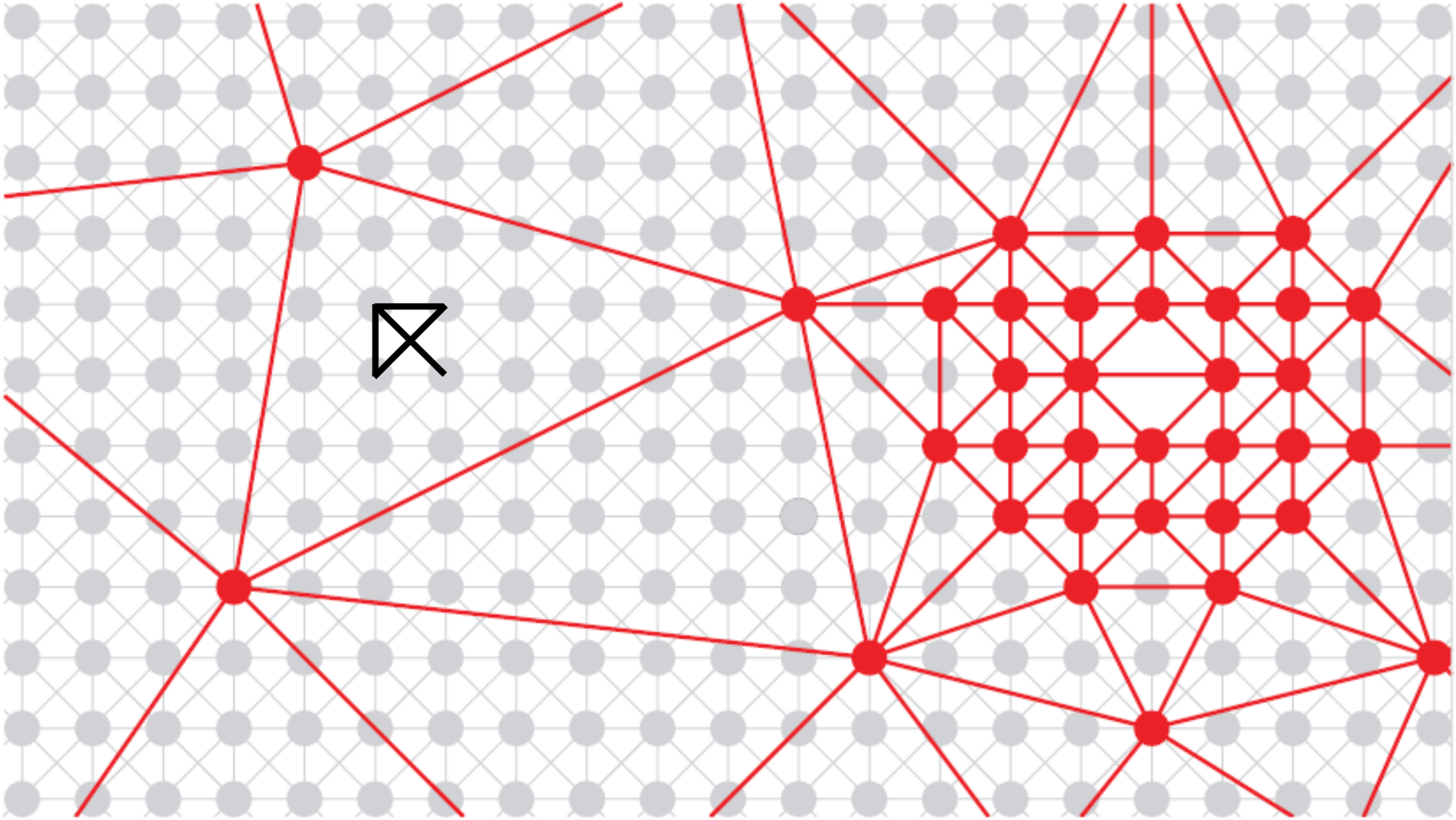
Integration for linear triangles



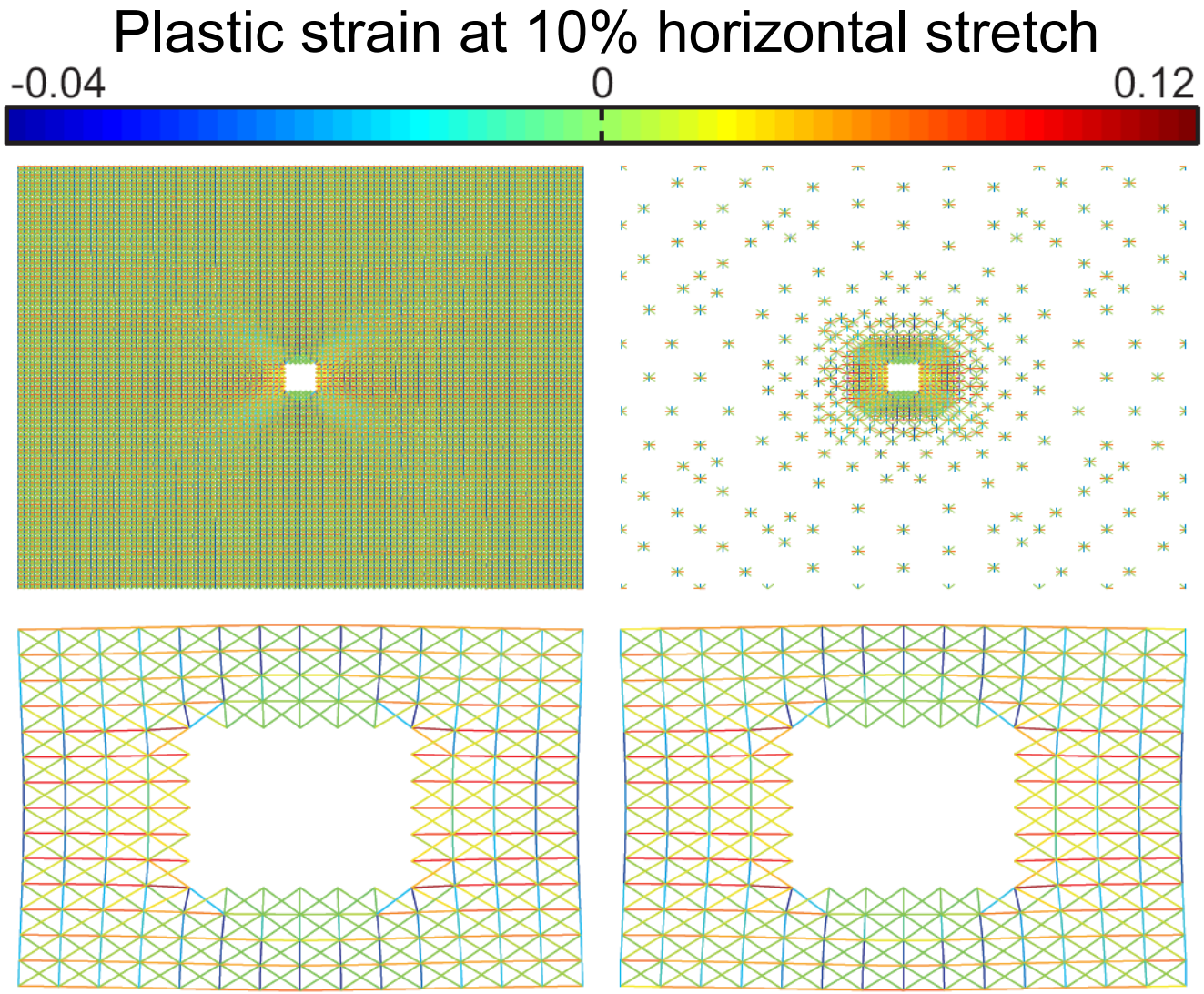
Integration 1 for linear triangles



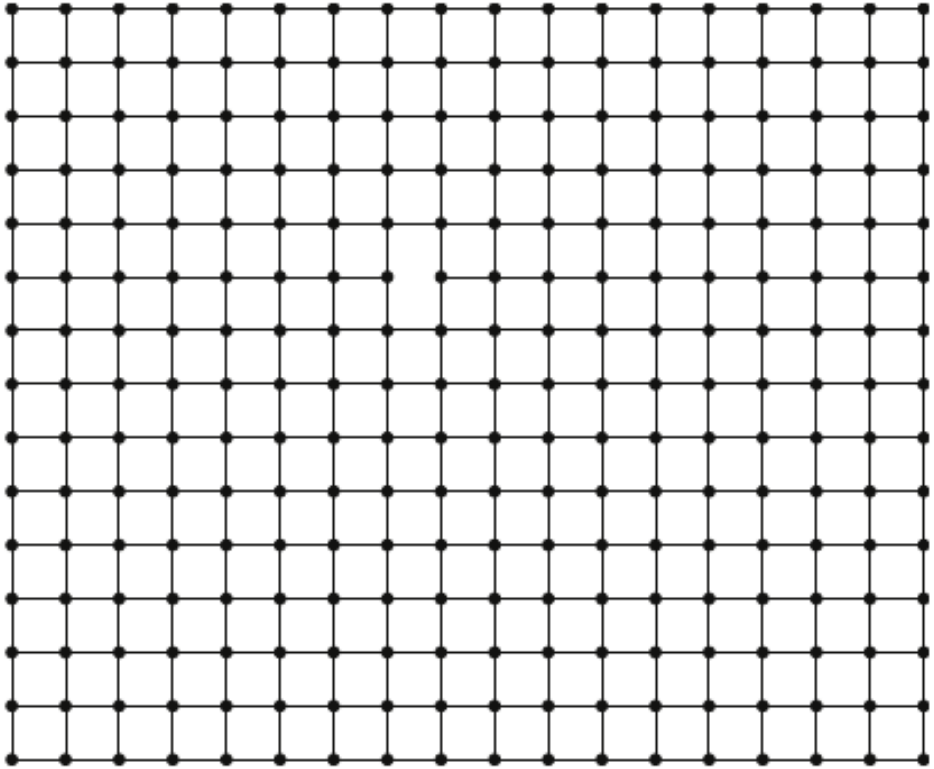
Integration 2 for linear triangles



Accuracy and efficiency



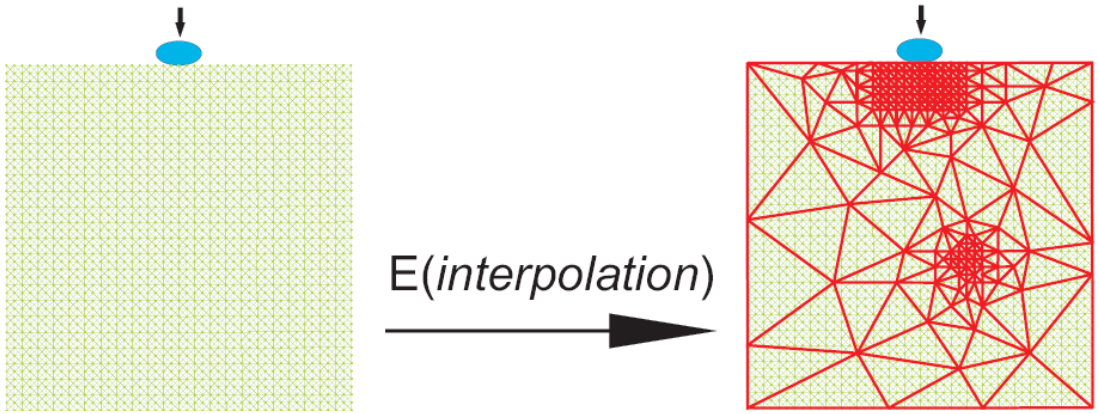
P. Kerfriden, S. Bordas



Euler Bernoulli beams:

- **Hermite interpolation in each beam**
- **nodal displacements**
- **nodal rotations**

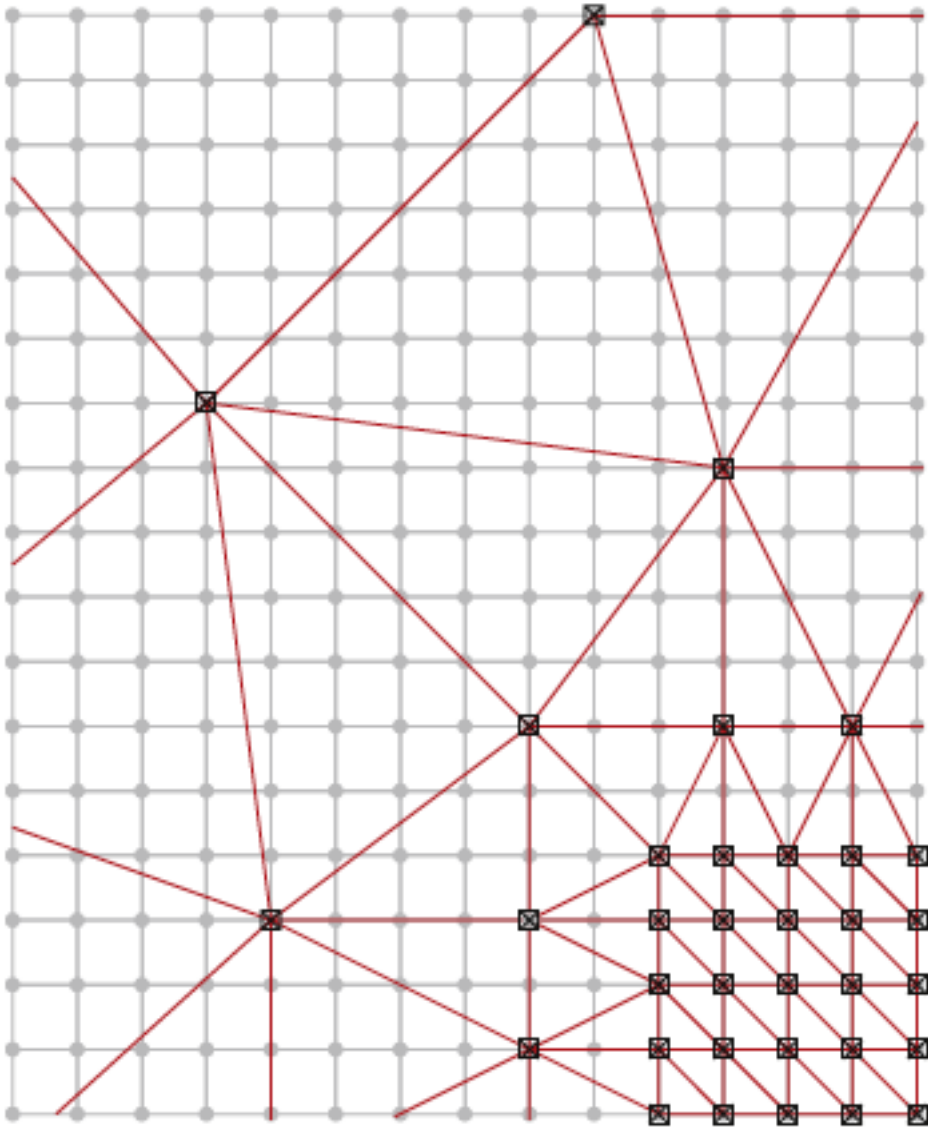
Interpolation



$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^T u \qquad E_{tot}(Nu_t) = \sum_{i=1}^{n} E_i(Nu_t) - f_{ex}^T Nu_t$$

$$u = Nu_t$$

QC method for planar beam lattice

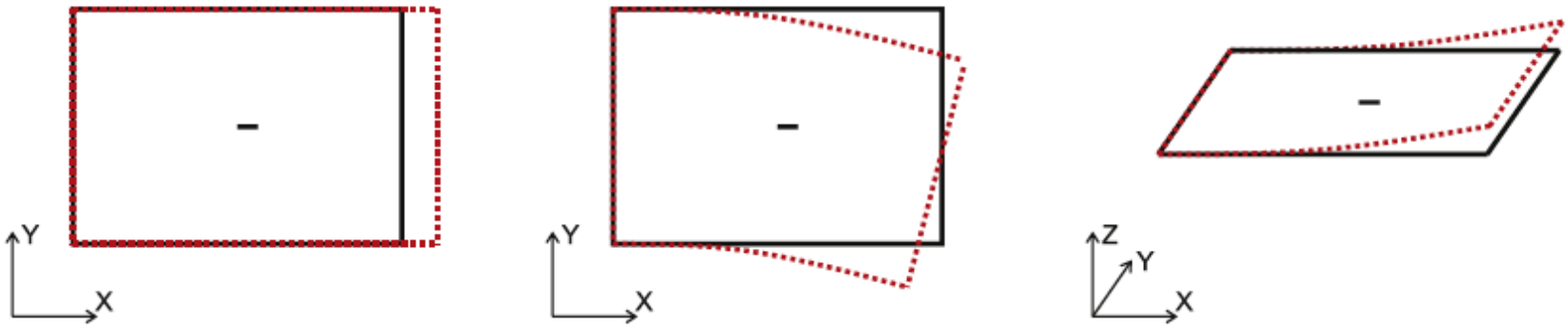


Nodal displacements: Linear

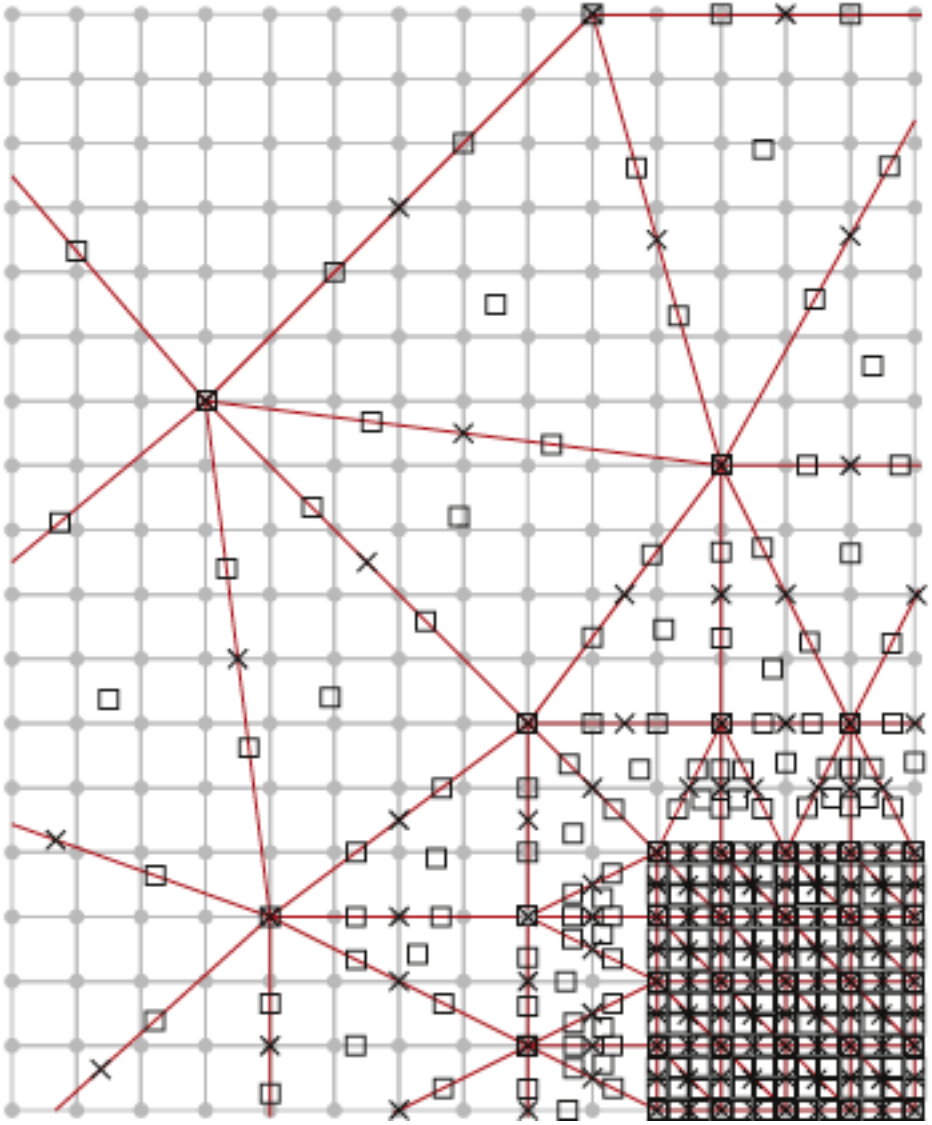
Nodal rotations: Linear

Conforming triangulations

Test cases



QC method for planar beam lattice

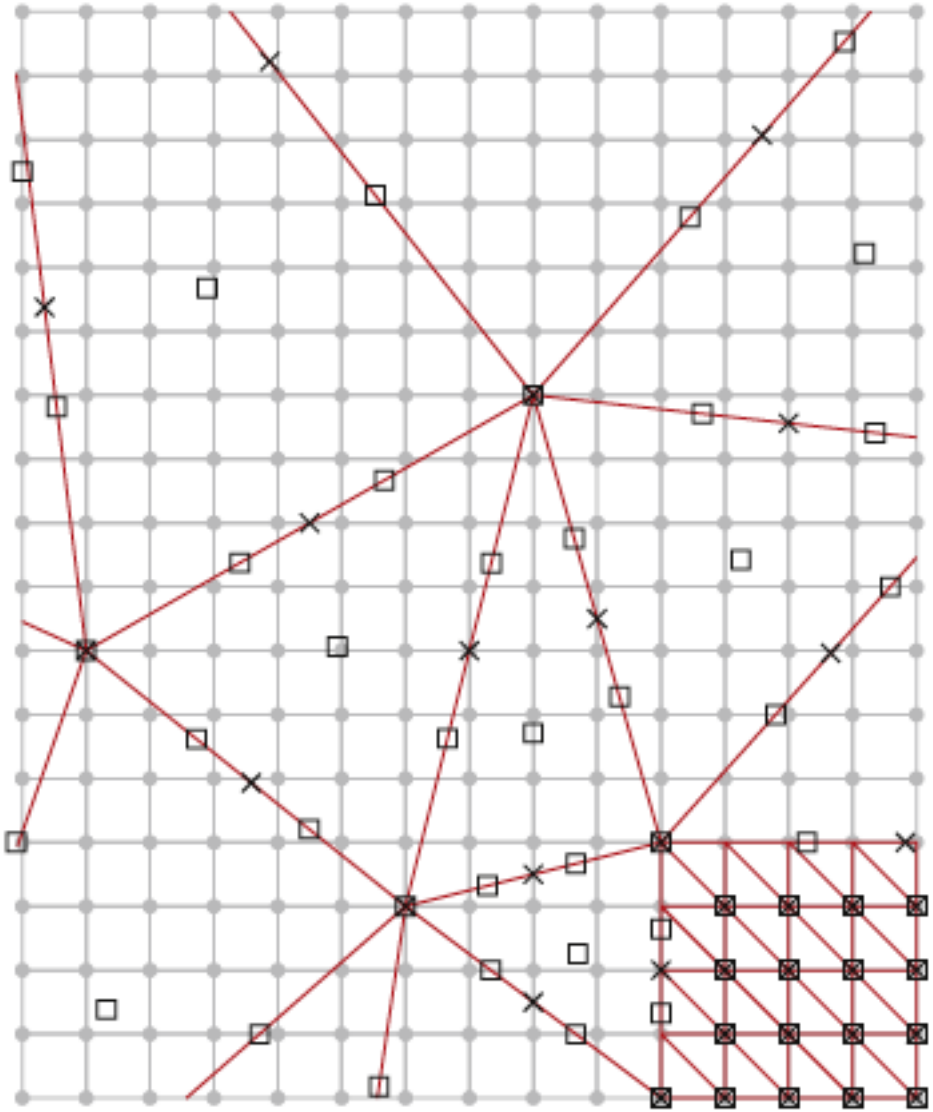


Nodal displacements: Cubic

Nodal rotations: Quadratic

Conforming triangulations

QC method for planar beam lattice

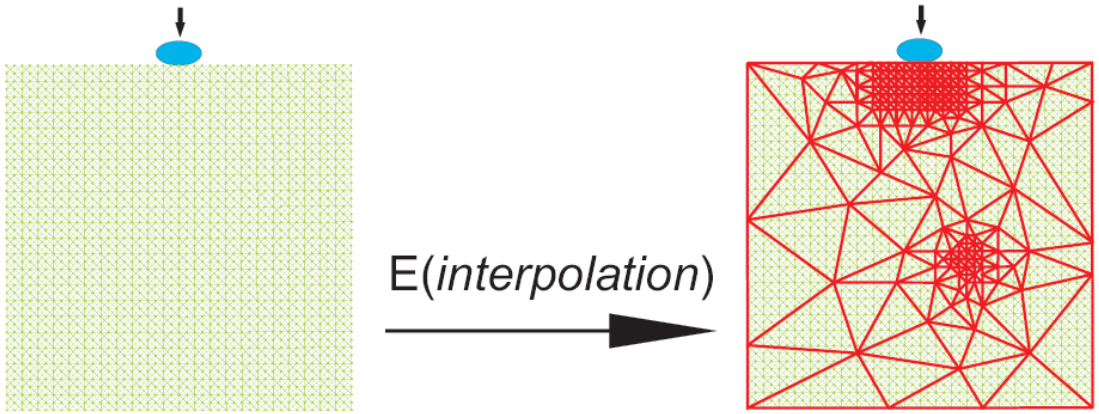


Nodal displacements: Cubic

Nodal rotations: Quadratic

Non-conforming triangulations

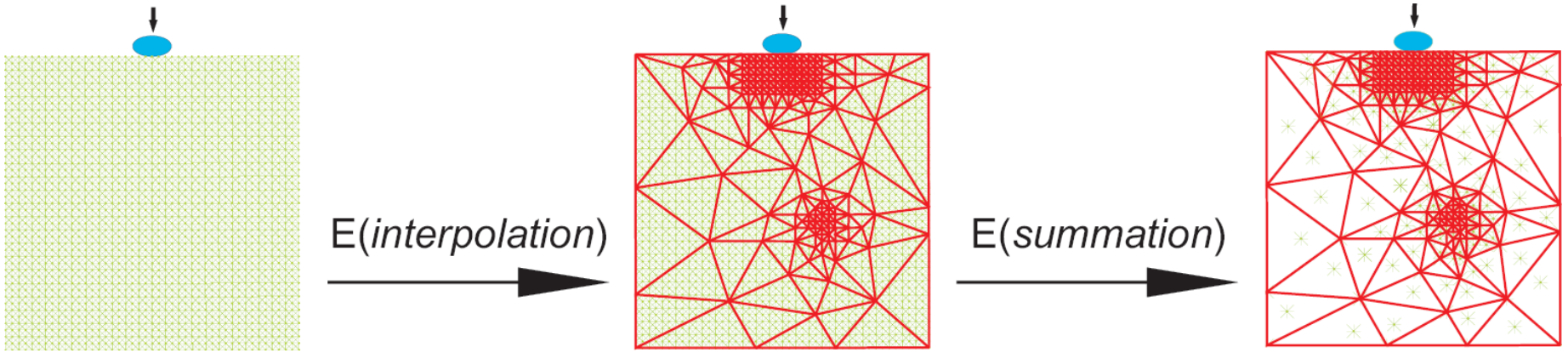
Interpolation



$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^T u \qquad E_{tot}(Nu_t) = \sum_{i=1}^{n} E_i(Nu_t) - f_{ex}^T Nu_t$$

$$u = Nu_t$$

Integration

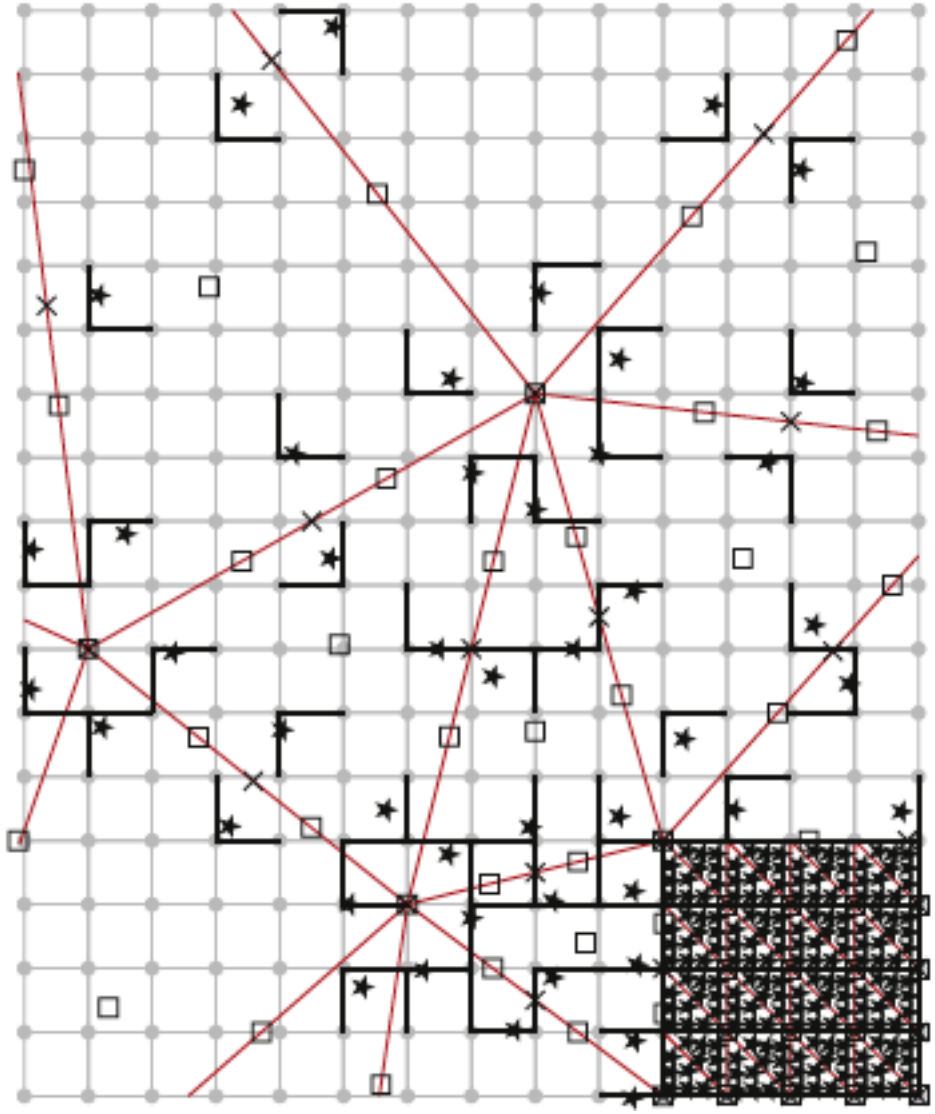


$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^T u \qquad E_{tot}(Nu_t) = \sum_{i=1}^{n} E_i(Nu_t) - f_{ex}^T Nu_t \qquad E_{tot}(Nu_t) = \sum_{i=1}^{s} E_i(Nu_t) - f_{ex}^T Nu_t$$

$$u = Nu_t$$

$$\sum_{i=1}^{n} E_i(Nu_t) \approx \sum_{i=1}^{s} w_i E_i(Nu_t)$$

QC method for planar beam lattice



Sampling beams near

Gauss points

Outline

1. Applications: discrete materials & discrete models

2. The Quasicontinuum method:

   interpolation & integration

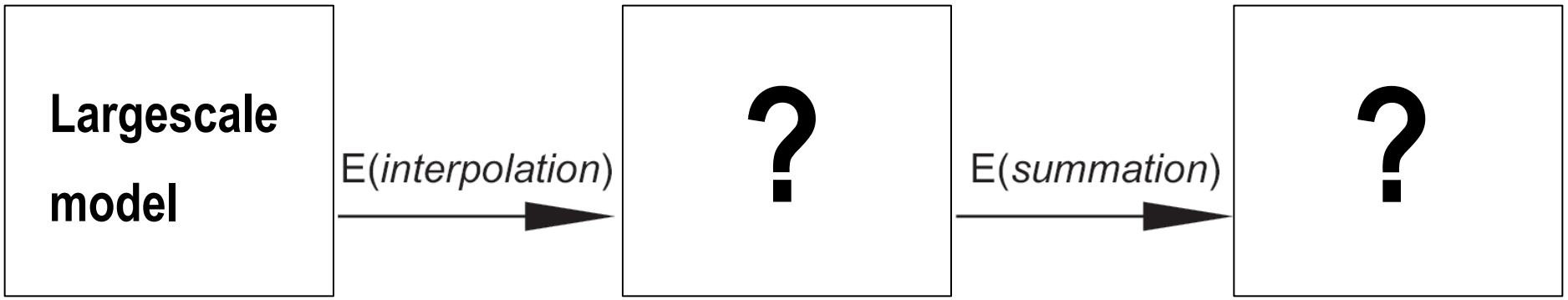
3. POD-based reduction methods:

   interpolation & integration

4. Concluding remarks

POD-based ROM (Ladevèze, Farhat, Chinesta,…)

E. Schenone, J.S. Hale, S. Bordas



$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^T u$$

$$E_{tot}(Nu_t) = \sum_{i=1}^{n} E_i(Nu_t) - f_{ex}^T Nu_t$$

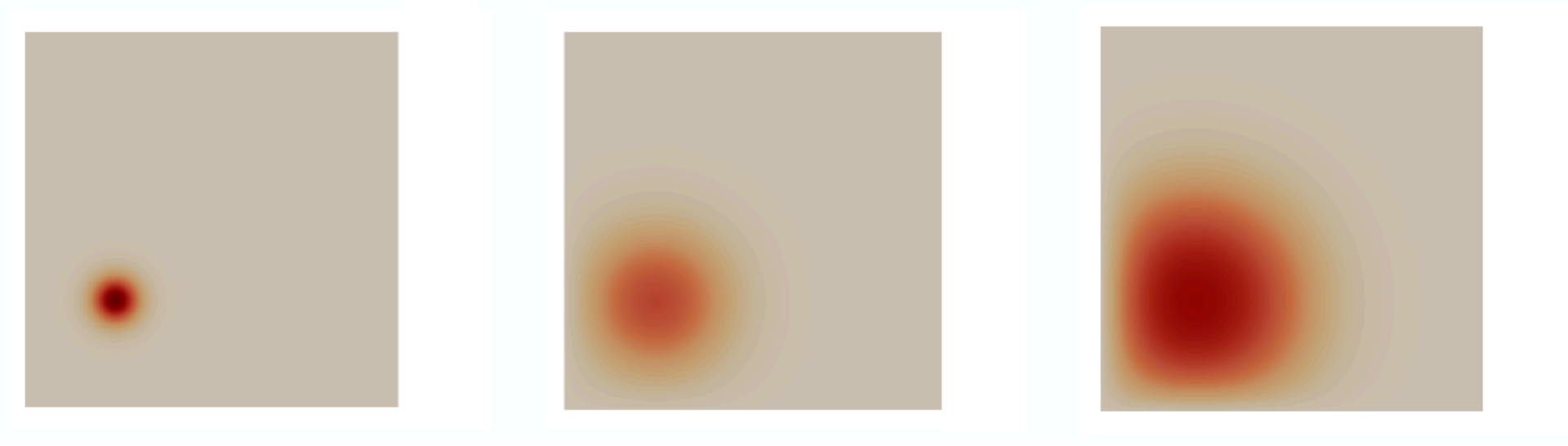
$$E_{tot}(Nu_t) = \sum_{i=1}^{s} E_i(Nu_t) - f_{ex}^T Nu_t$$

$$u = Nu_t$$

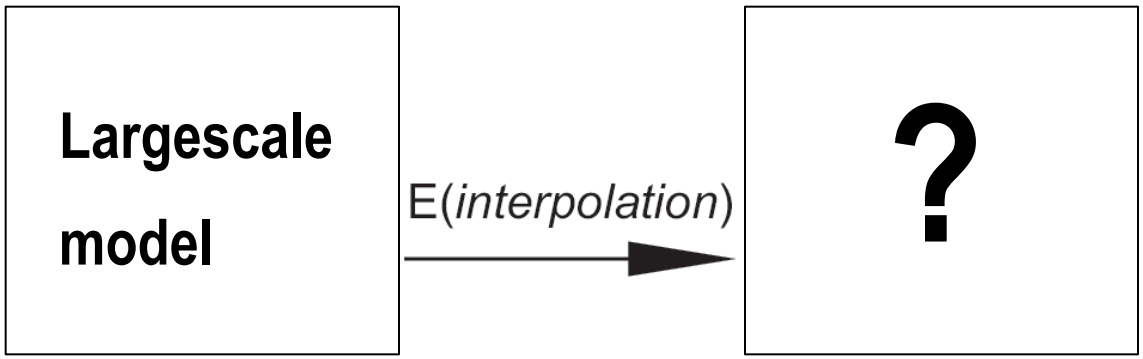
$$\sum_{i=1}^{n} E_i(Nu_t) \approx \sum_{i=1}^{s} w_i E_i(Nu_t)$$

Nonlinear reaction diffusion

$$K\Delta u = f(u)$$



POD-based ROM (Ladevèze, Farhat, Chinesta,…)



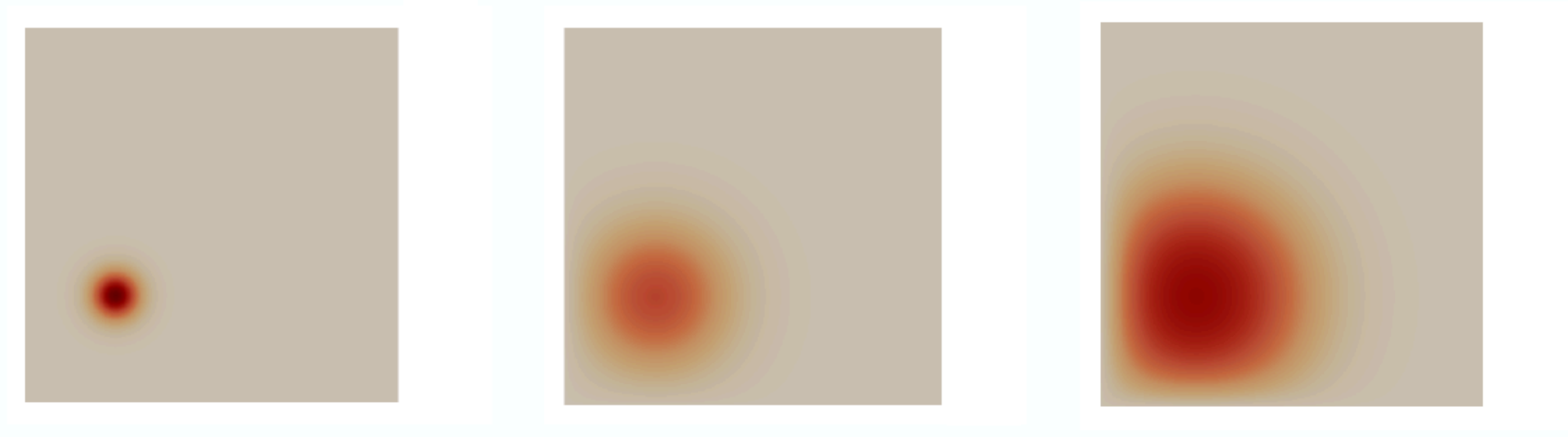
$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^T u \qquad E_{tot}(Nu_t) = \sum_{i=1}^{n} E_i(Nu_t) - f_{ex}^T Nu_t$$

$$u = Nu_t$$

How to find N: *Offline snapshot/training generation*

$$K\Delta u = f(u)$$

many results: $u_1, u_2, \ldots, u_{150}$

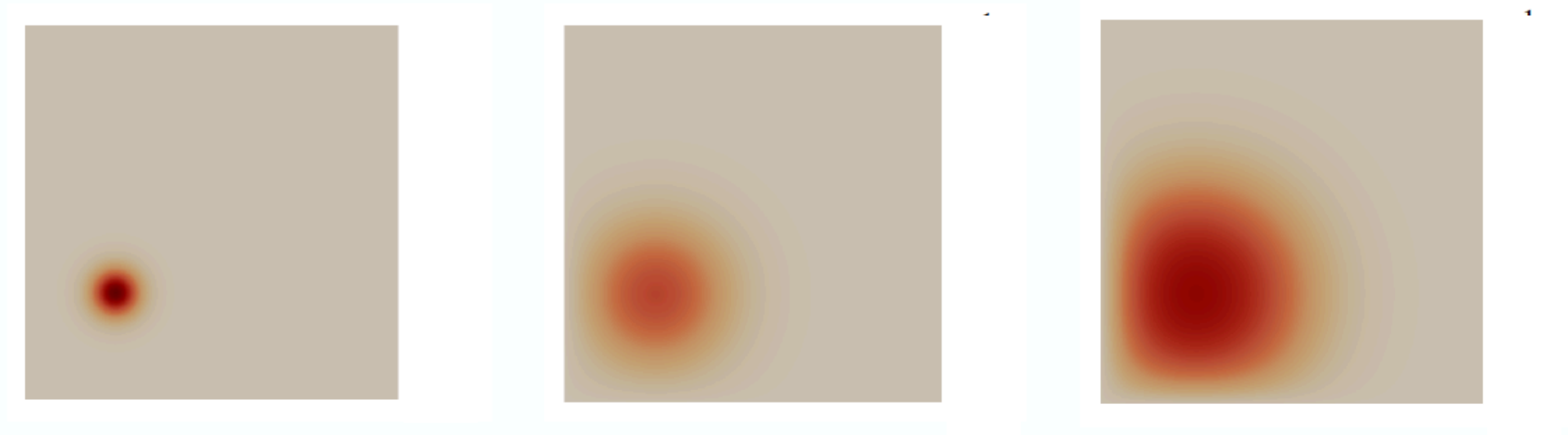


time

How to find N: *Offline snapshot/training generation*

$$K\Delta u = f(u) \xrightarrow{\text{many results: } u_1, u_2, \ldots, u_{150}} N = [u_1, u_2, \ldots, u_{150}]$$

reduced basis



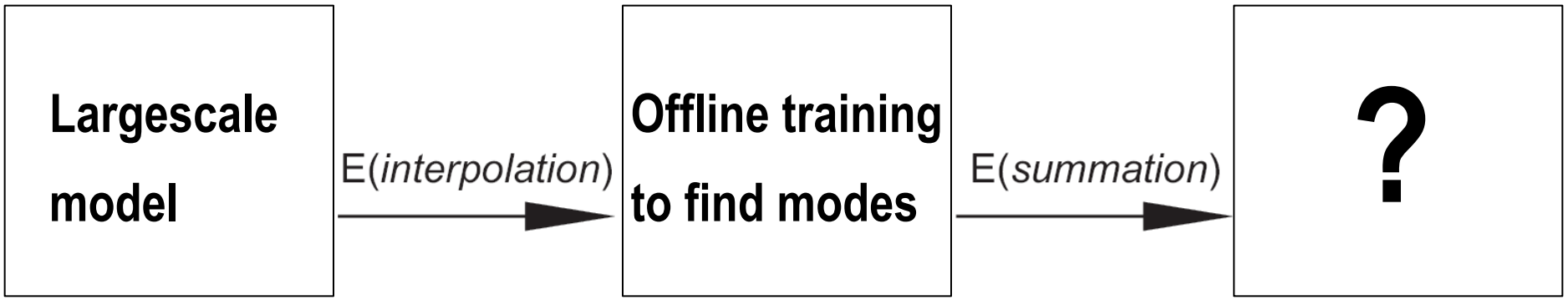
time →

Or apply first singular value decomposition to $N = [u_1, u_2, \ldots, u_{150}]$

and use the modes with the largest eigenvalues $N = [\varphi_1, \varphi_2, \ldots, \varphi_{30}]$

Note: N is full

POD-based ROM



$$E_{tot}(u) = \sum_{i=1}^{n} E_i(u) - f_{ex}^T u$$

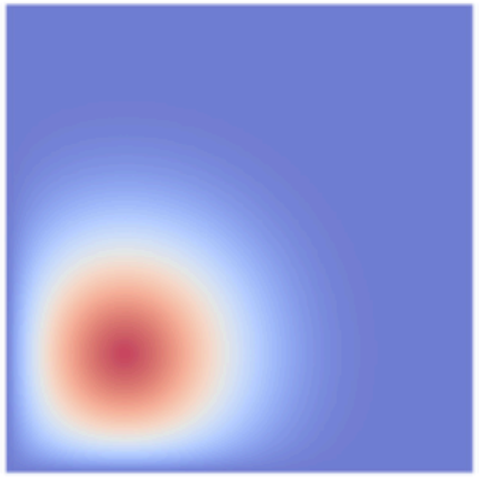
$$E_{tot}(Nu_t) = \sum_{i=1}^{n} E_i(Nu_t) - f_{ex}^T Nu_t$$

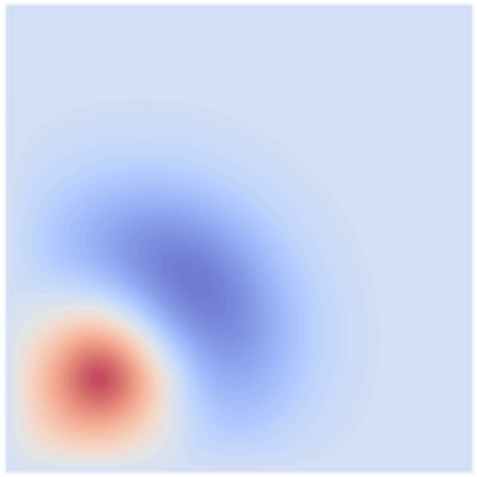
$$E_{tot}(Nu_t) = \sum_{i=1}^{s} E_i(Nu_t) - f_{ex}^T Nu_t$$

$$u = Nu_t$$

$$\sum_{i=1}^{n} E_i(Nu_t) \approx \sum_{i=1}^{s} w_i E_i(Nu_t)$$

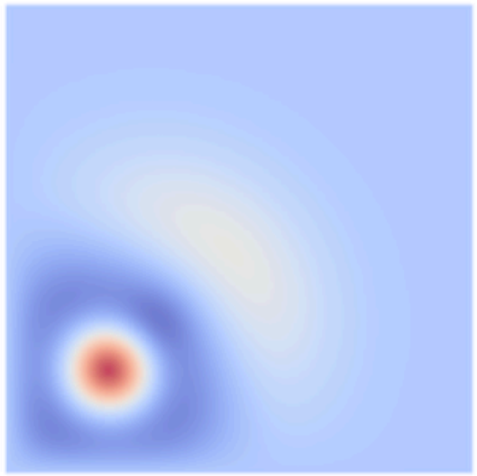
POD-based ROM

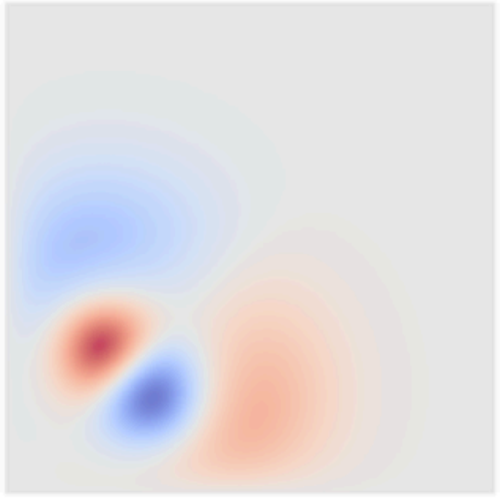
φ_1 

φ_2 

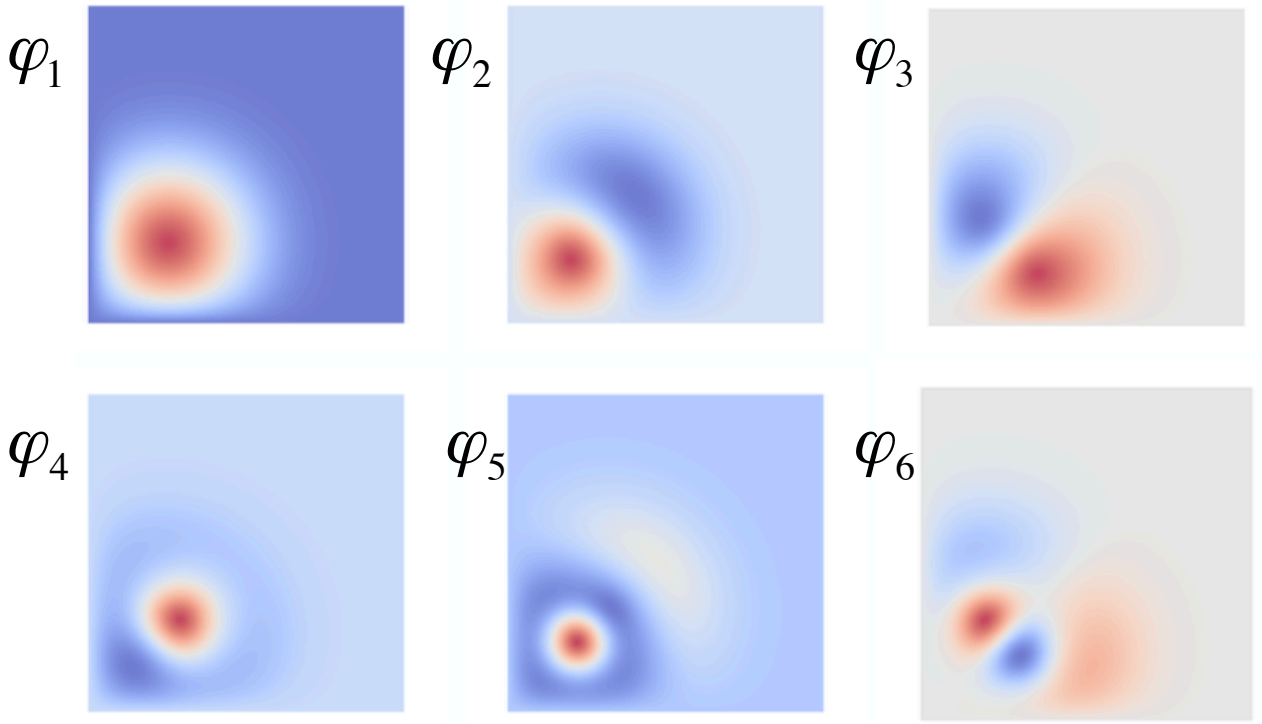
φ_3 

φ_4 

φ_5 

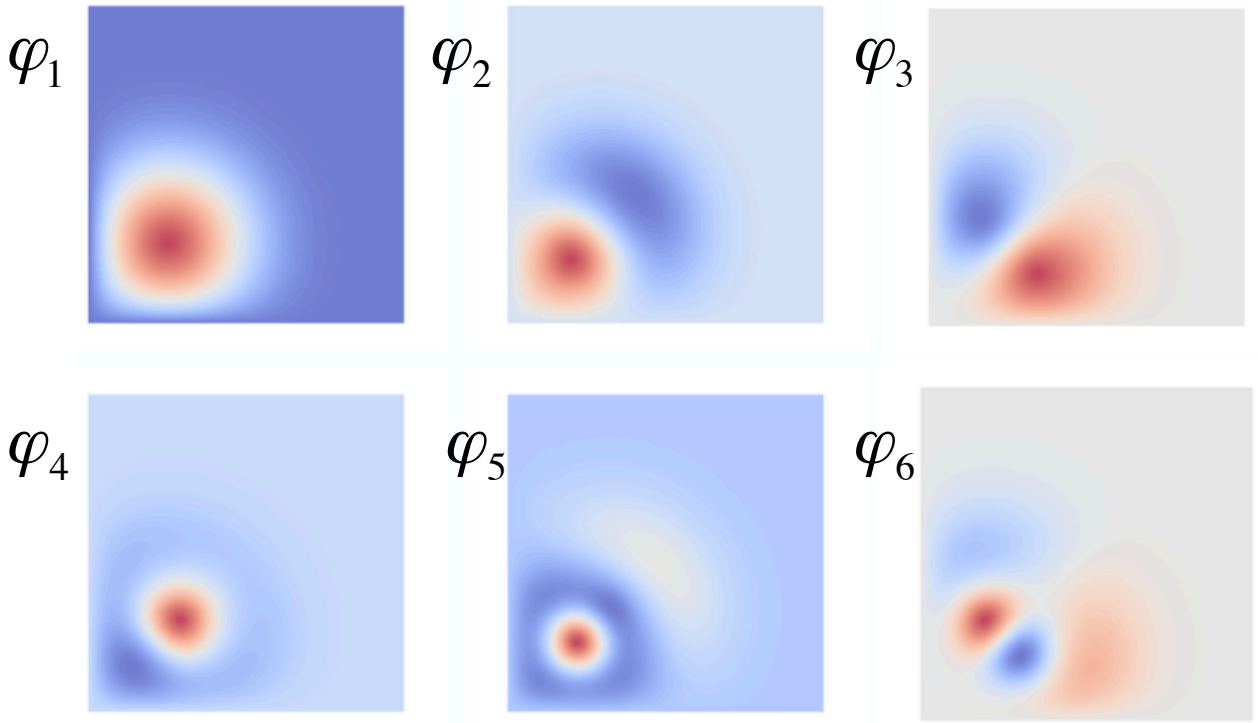
φ_6 

How to select reduced integration points for oscillatory basis functions?

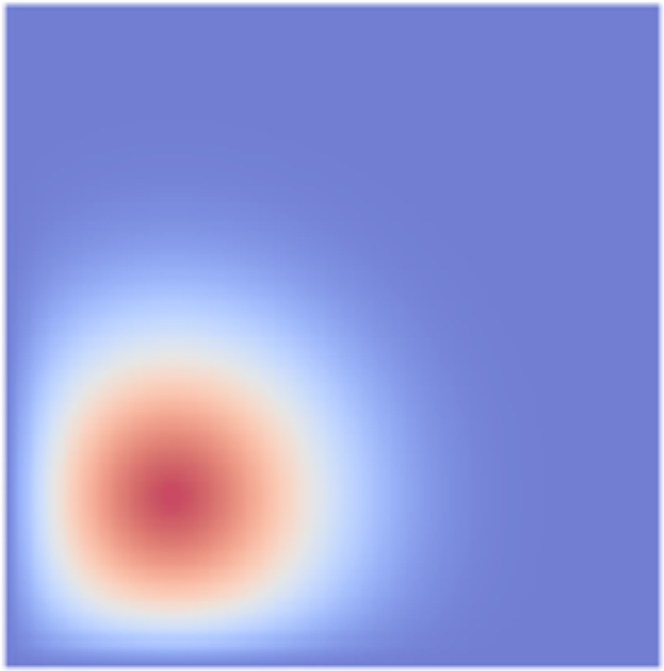


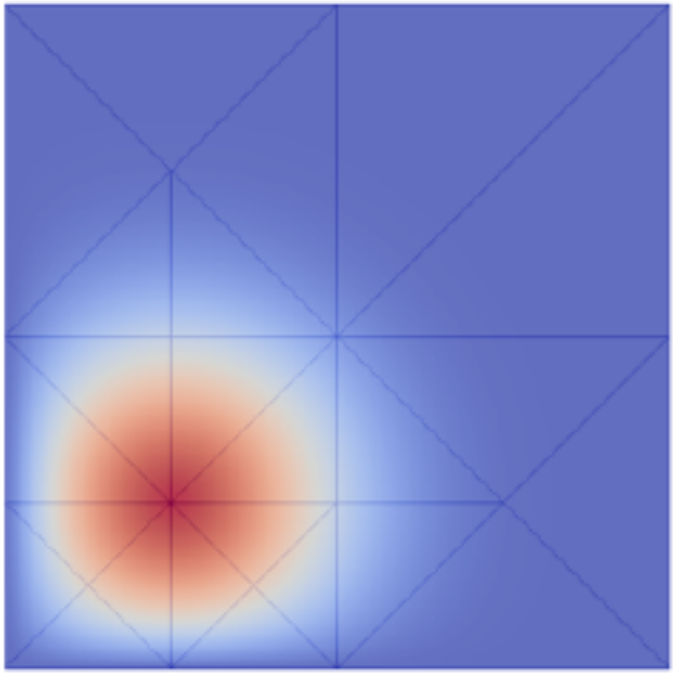
How to select reduced integration points for oscillatory basis functions?

Locally approximate the modes linearly!

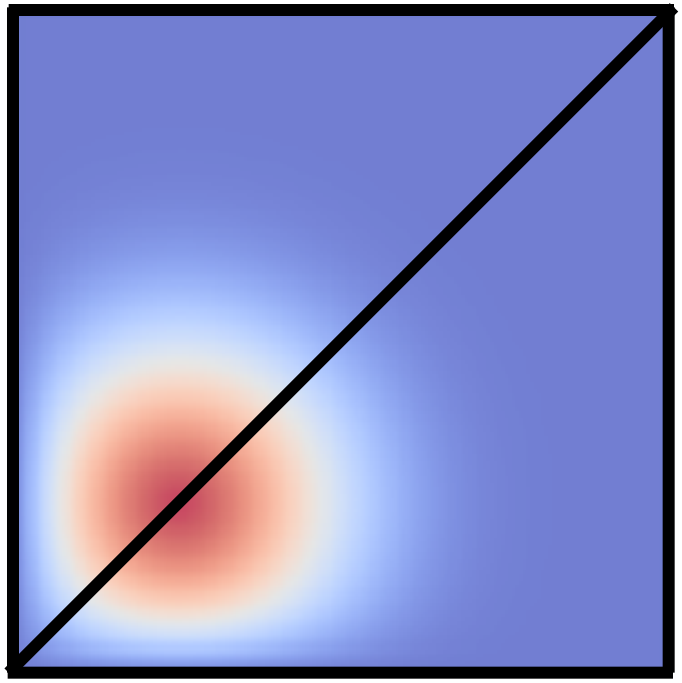


Let's look at φ_1



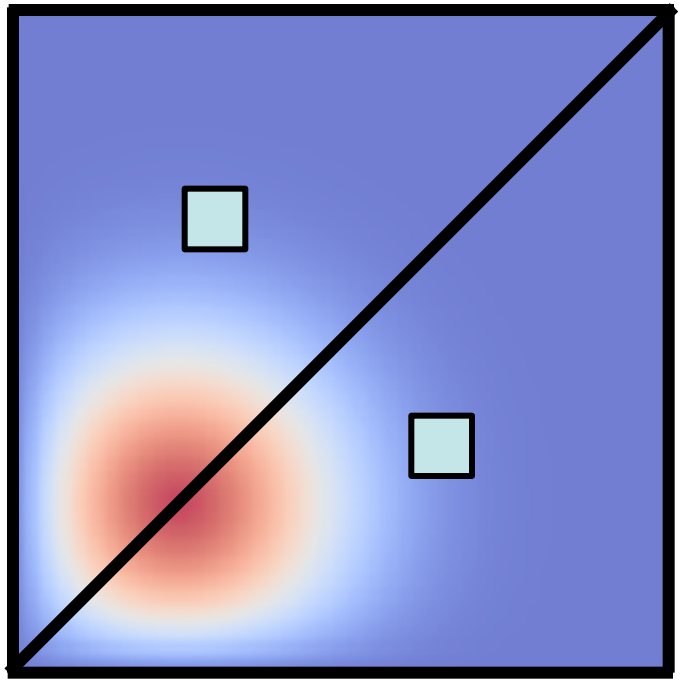


Let's look at φ_1

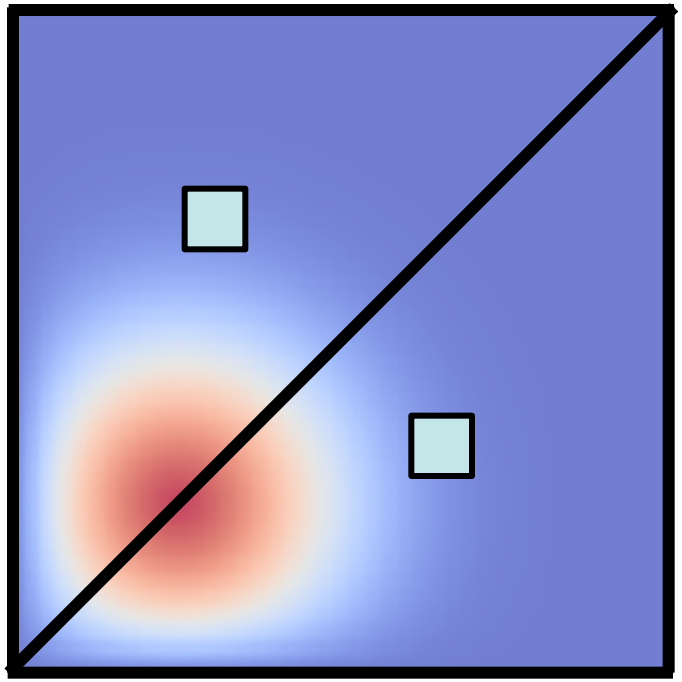


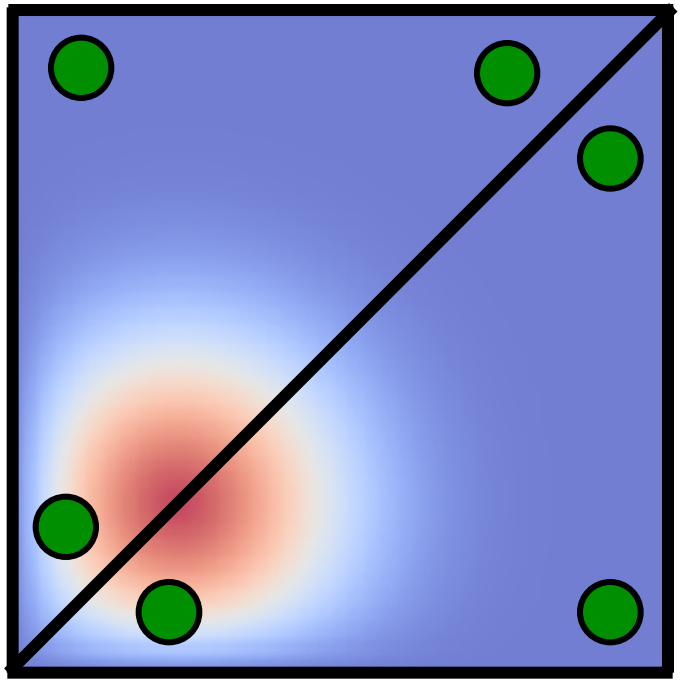
Start with coarse mesh

Let's look at φ_1

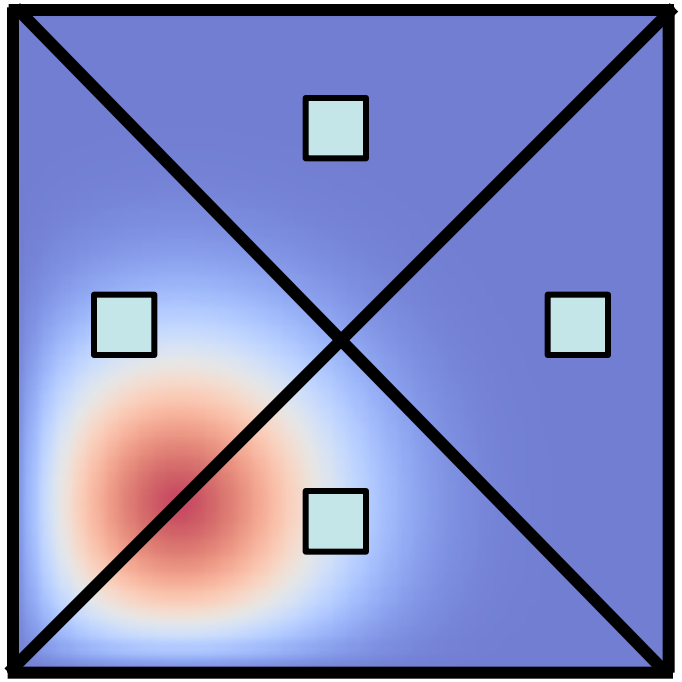


Let's look at φ_1

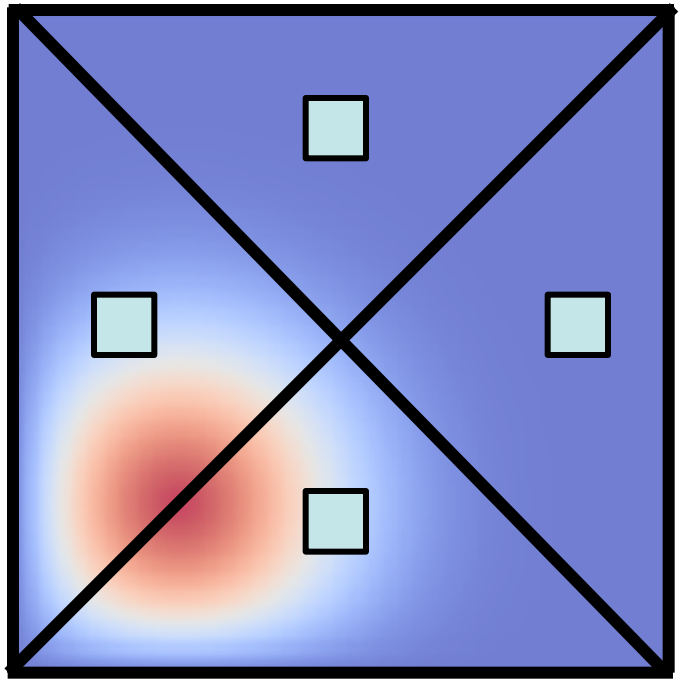


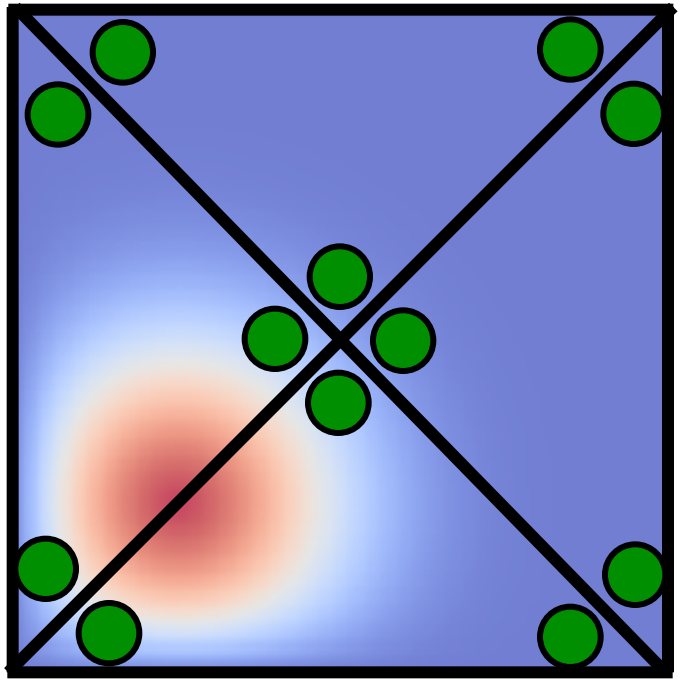


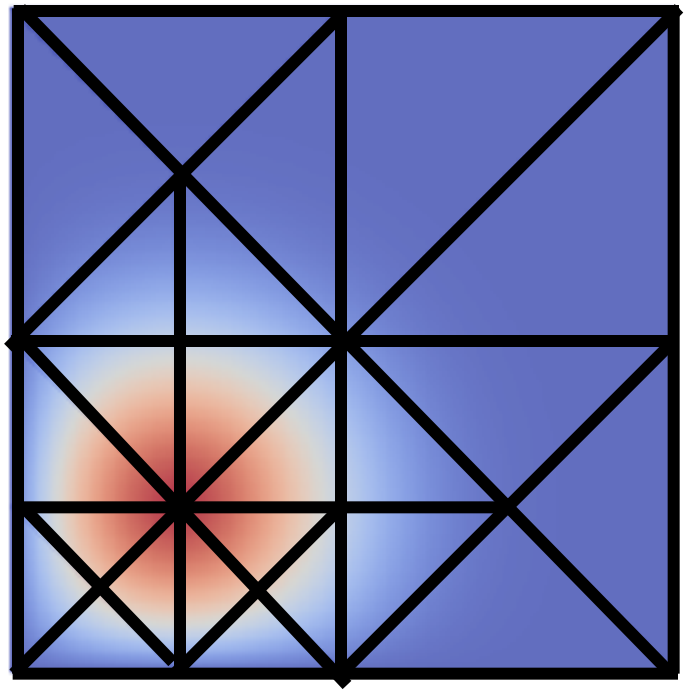
Let's look at φ_1



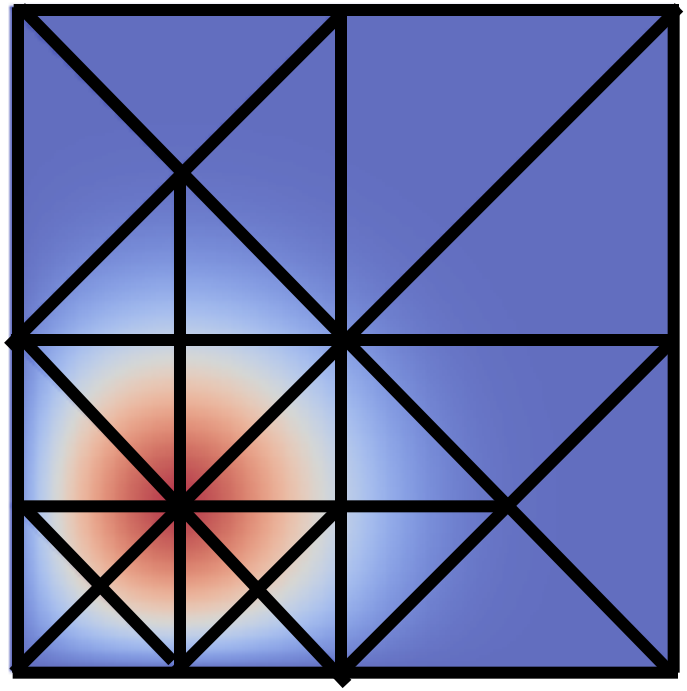
Let's look at φ_1

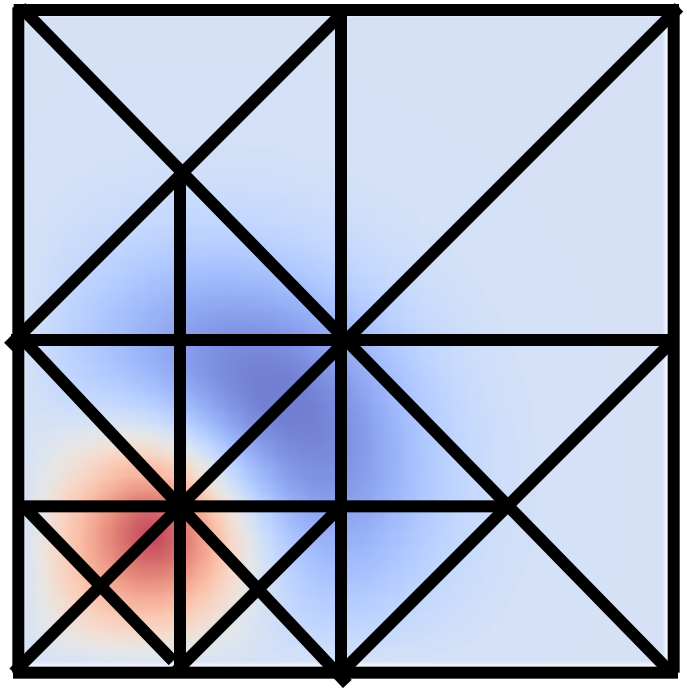






Final mesh of φ_1



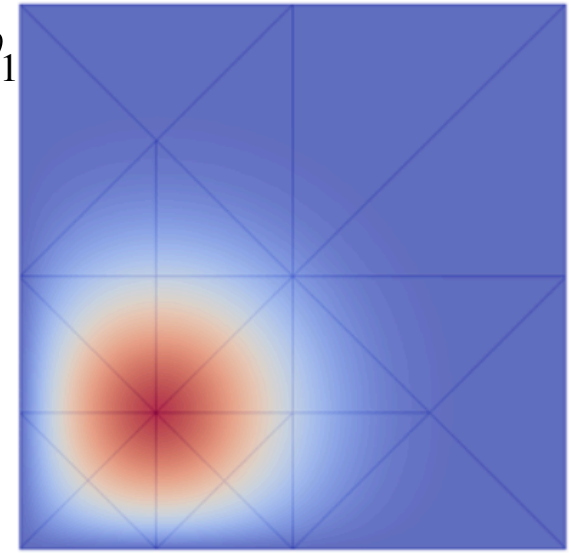


Final mesh of φ_1

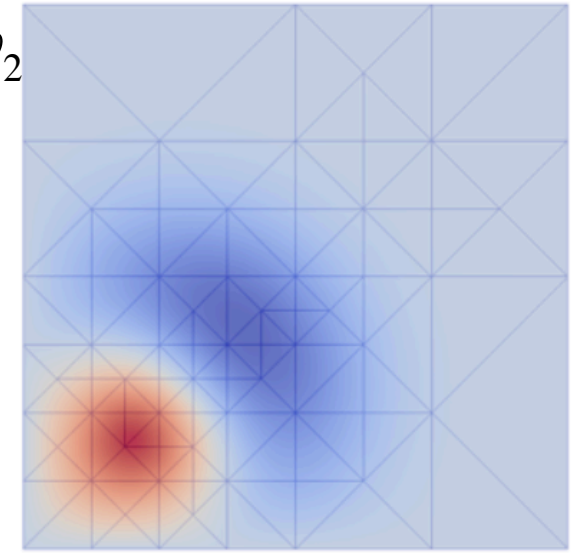
is the first mesh for φ_2

POD-based ROM

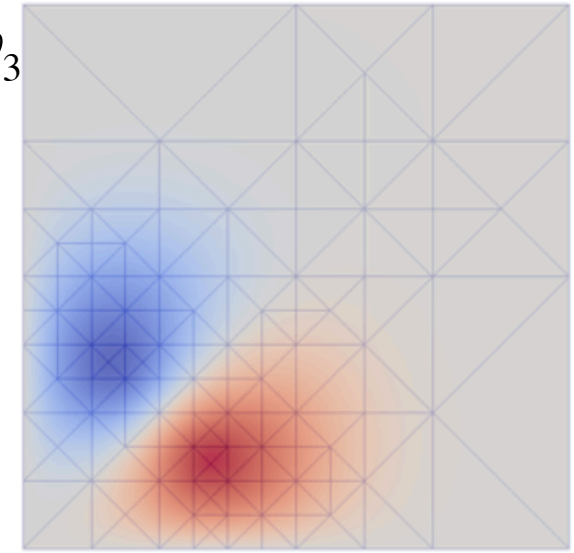
φ_1



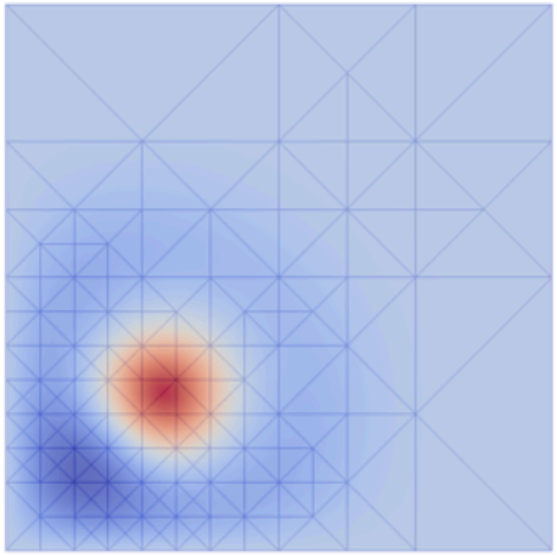
φ_2



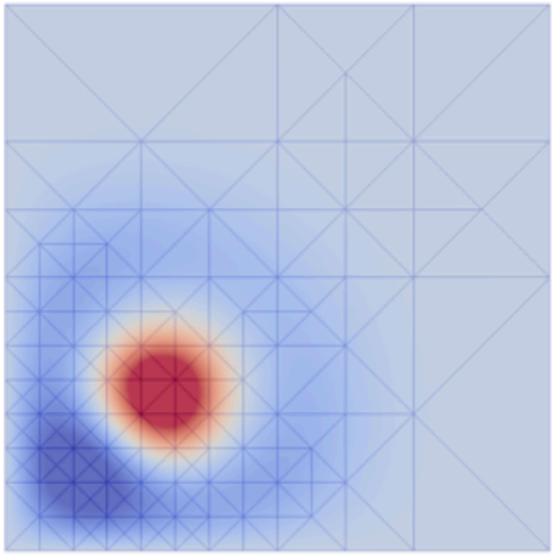
φ_3



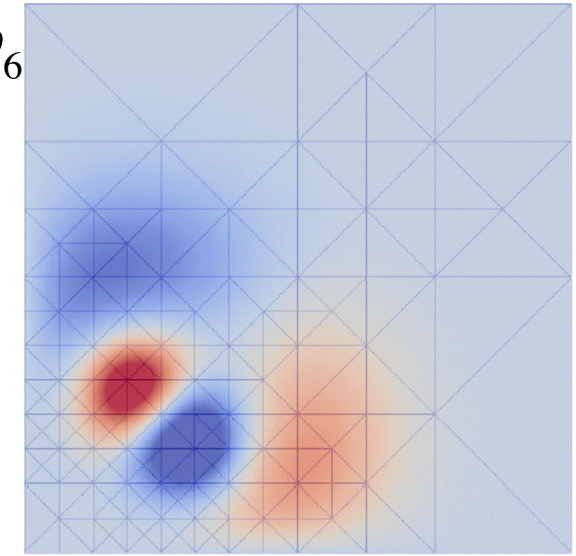
φ_4



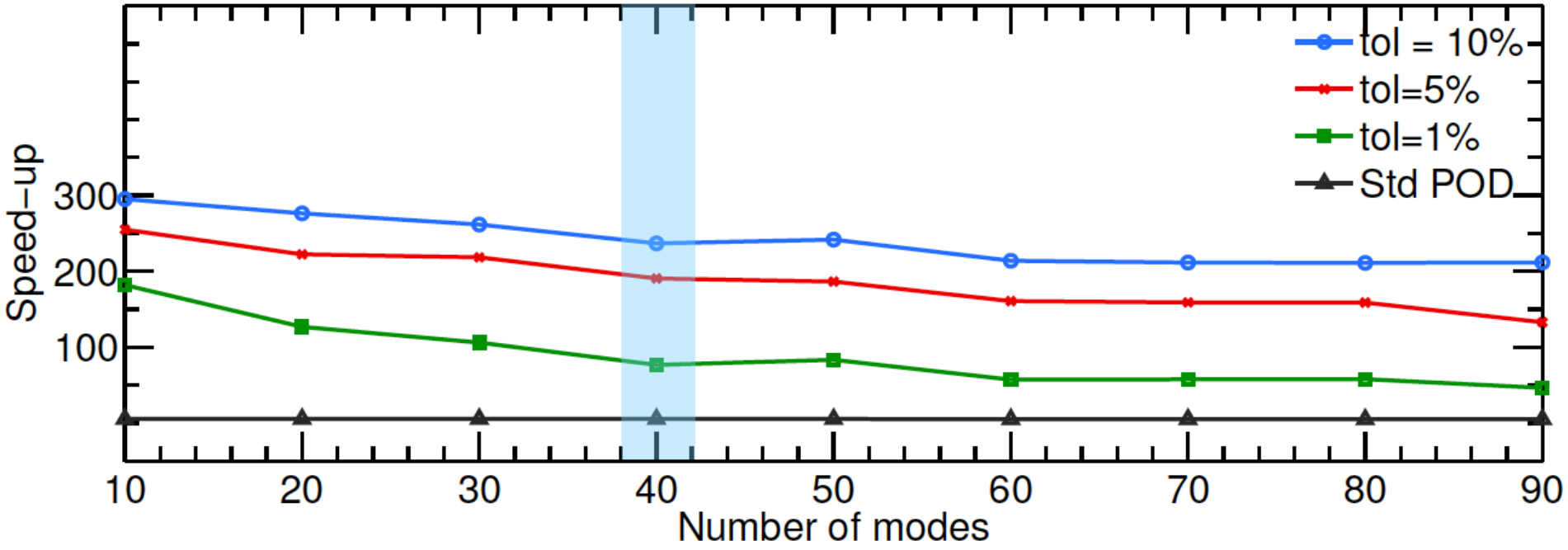
φ_5



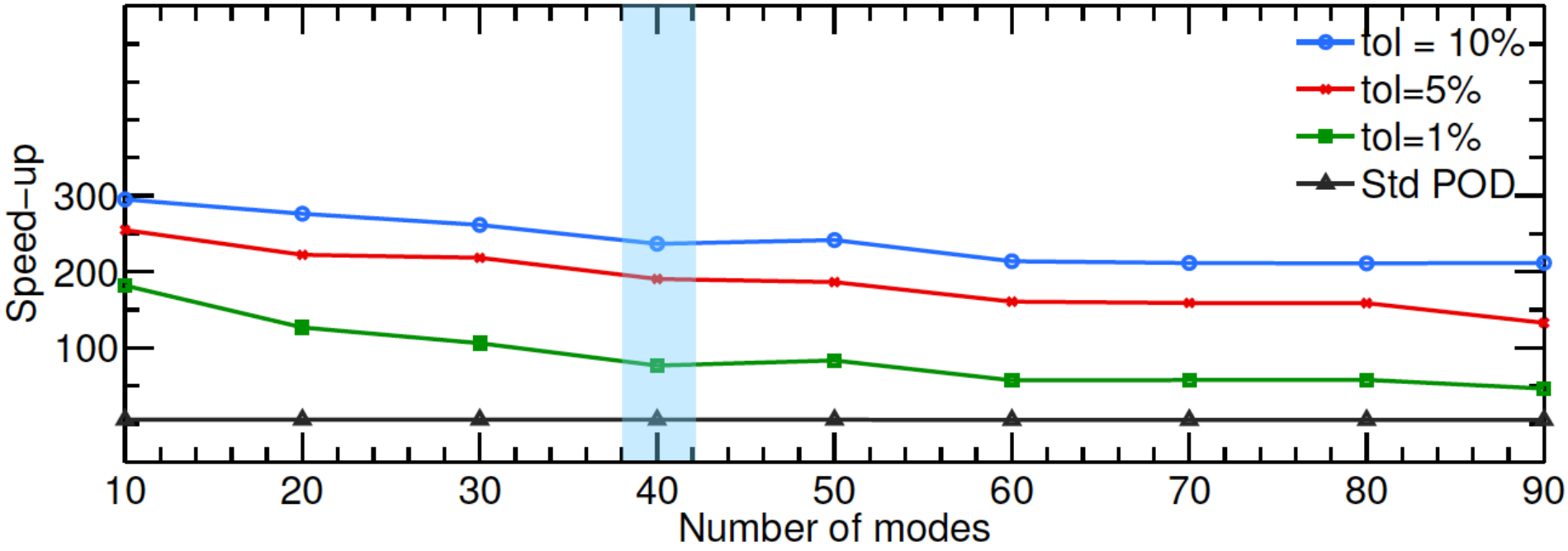
φ_6

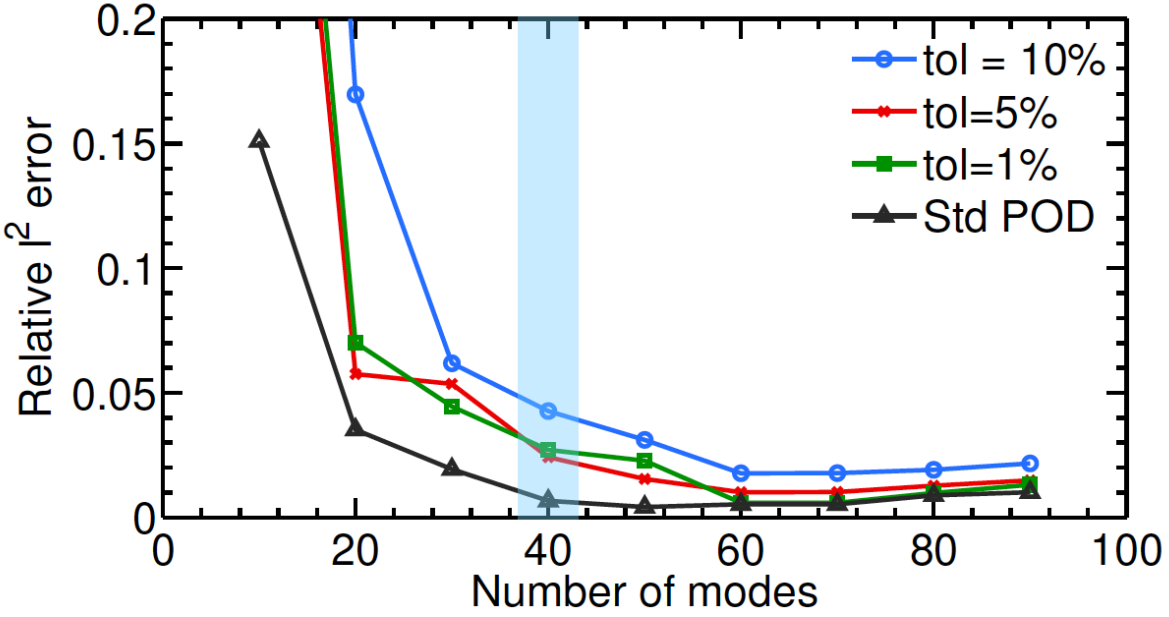


POD-based ROM



POD-based ROM



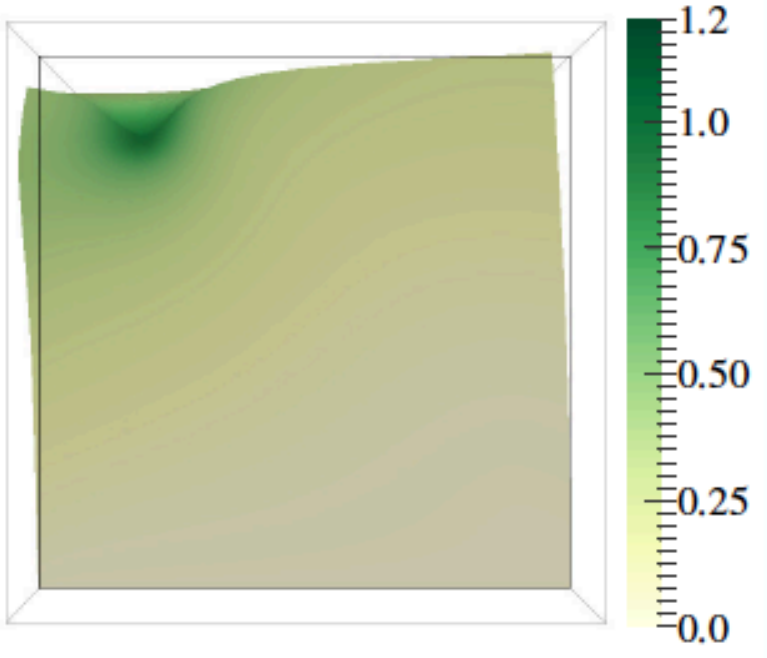


RUES | RESEARCH UNIT IN ENGINEERING SCIENCES

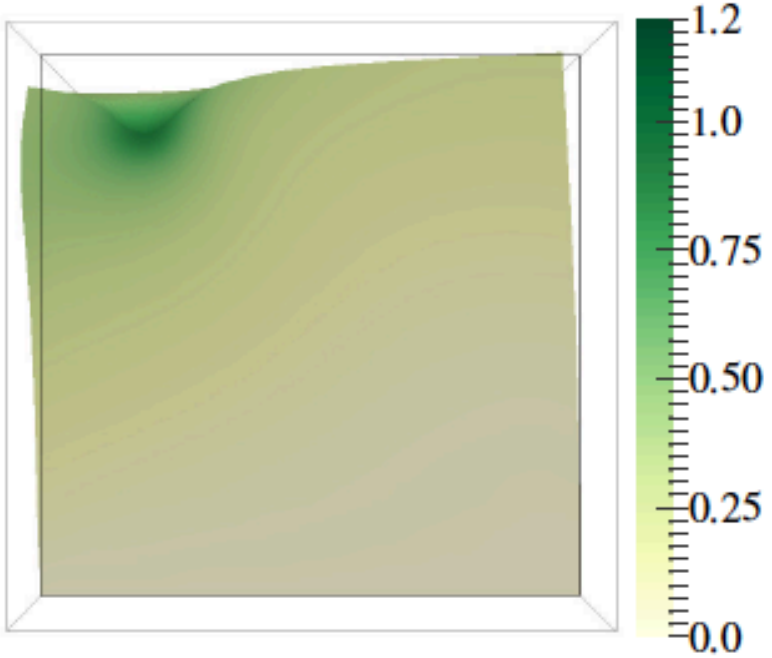
UNIVERSITÉ DU LUXEMBOURG

Indentation of a hyperelastic cube

30x faster than standard POD



FEM



Reduced integrated POD

Concluding remarks

	QC method	POD-based ROM
Offline training to find *N*	None	A LOT, so only useful for optimisation

Concluding remarks

	QC method	POD-based ROM
Offline training to find *N*	None	A LOT, so only useful for optimisation
Finding integration points	Every time the same	Changes every time

Concluding remarks

	QC method	POD-based ROM
Offline training to find *N*	None	A LOT, so only useful for optimisation
Finding integration points	Every time the same	Changes every time
Localized behavior	Fully resolved regions	Limited… not easy

Concluding remarks

	QC method	POD-based ROM
Offline training to find *N*	None	A LOT, so only useful for optimisation
Finding integration points	Every time the same	Changes every time
Localized behavior	Fully resolved regions	Limited… not easy
Adaptivity of *N*	Simple, borrow from FE	Difficult I guess..

Concluding remarks

	QC method	POD-based ROM
Offline training to find *N*	None	A LOT, so only useful for optimisation
Finding integration points	Every time the same	Changes every time
Localized behavior	Fully resolved regions	Limited… not easy
Adaptivity of *N*	Simple, borrow from FE	Difficult I guess..
Type of discretisation	**ONLY REGULAR**	**ANY**

Ongoing & future work

1. QC for irregular networks

2. (goal-oriented) Adaptivity for QC

3. Apply QC to true materials, no academic examples

4. Geometrical scaling of POD-modes

5. Coupling of POD domains